

TUGAS AKHIR - KS141501

**PENGEMBANGAN SISTEM RESERVASI KENDARAAN
DENGAN MENGGUNAKAN PENDEKATAN
MANAJEMEN PROSES BISNIS (STUDI KASUS :
PT.PERTAMINA EP)**

**JOSHUA SATRIAWAN
NRP 5212 100 137**

**Dosen Pembimbing
Arif Wibisono, S.Kom., M.Sc**

**JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2016**

FINAL PROJECT - KS 141501

***THE DEVELOPMENT OF VEHICLE RESERVATION
SYSTEMS USING BUSINESS PROCESS MANAGEMENT
APPROACH (CASE STUDY : PT.PERTAMINA EP)***

**JOSHUA SATRIAWAN
NRP 5212 100 137**

**Supervisor
Arif Wibisono, S.Kom., M.Sc**

**INFORMATION SYSTEMS DEPARTMENT
Information Technology Faculty
Sepuluh Nopember Institut of Technology
Surabaya 2016**

LEMBAR PENGESAHAN

PENGEMBANGAN SISTEM RESERVASI KENDARAAN DENGAN MENGGUNAKAN PENDEKATAN MANAJEMEN PROSES BISNIS (STUDI KASUS : PT.PERTAMINA EP)

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

JOSHUA SATRIAWAN
NRP. 5212 100 137

Surabaya, Juli 2016

**KETUA
JURUSAN SISTEM INFORMASI**

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP.19650310 199102 1 001

LEMBAR PERSETUJUAN

PENGEMBANGAN SISTEM RESERVASI KENDARAAN DENGAN MENGGUNAKAN PENDEKATAN MANAJEMEN PROSES BISNIS (STUDI KASUS : PT.PERTAMINA EP)

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

JOSHUA SATRIAWAN

NRP. 5212 100 137

Disetujui Tim Penguji : Tanggal Ujian : April 2016
Periode Wisuda : September 2016

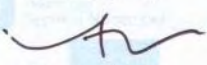
Arif Wibisono, S.Kom., M.Sc.


(Pembimbing I)

Mahendrawathi ER, S.T., M.Sc., Ph.D


(Penguji I)

Amna Shifia Nisafani, S.Kom, M.Sc.


(Penguji II)

**PENGEMBANGAN SISTEM RESERVASI
KENDARAAN DENGAN MENGGUNAKAN
PENDEKATAN MANAJEMEN PROSES BISNIS (STUDI
KASUS : PT.PERTAMINA EP)**

Nama Mahasiswa : JOSHUA SATRIAWAN
NRP : 5212 100 137
Jurusan : SISTEM INFORMASI FTIF-ITS
Dosen Pembimbing 1 : Arif Wibisono, S.Kom., M.Sc

ABSTRAK

Sistem reservasi kendaraan sangat penting bagi perusahaan manapun yang berfungsi agar karyawan perusahaan dapat menggunakan kendaraan milik kantor untuk keperluan bisnis kantor. Namun terdapat beberapa kendala dalam sistem tersebut yaitu jumlah kendaraan dan karyawan yang sangat banyak yang mengandalkan sistem reservasi kendaraan ini, terlebih lagi tidak terdapat sebuah software atau aplikasi yang mendukung sistem ini.

Penelitian ini membahas mengenai pengembangan sistem reservasi kendaraan (vehicle reservation system) menggunakan pendekatan process-oriented. Pada aplikasi ini sistem akan bekerja secara online dan menghubungkan antara user (orang yang melakukan request) dan approvers (orang yang menanggapi request / memberikan feedback). Alasan mengapa penelitian ini membahas menggunakan metode process-oriented karena aplikasi vehicle reservation system adalah sebuah application to approval process. Selanjutnya disiplin ilmu yang digunakan adalah business management process (BPM), disiplin ilmu ini digunakan untuk mengembangkan aplikasi ini karena process instance yang terdapat dalam BPM tersebut sangat jelas dijelaskan dan cocok untuk diterapkan pada studi kasus ini.

Hasil pada penelitian ini adalah sebuah aplikasi berbasis web vehicle reservation system yang dikembangkan menggunakan Bonitasoft, salah satu BPM software yang diakui di dunia, dengan menggunakan studi kasus PT.Pertamina EP yang merupakan salah satu perusahaan minyak terbesar di Indonesia. Aplikasi ini dapat menghubungkan seluruh actor yang terlibat dalam proses bisnis ini, sehingga permasalahan mengenai reservasi peminjaman kendaraan akan teratasi. Dengan adanya aplikasi ini maka proses yang berjalan dapat ter-automatisasi sehingga actor yang menjalankan proses ini menjadi terfasilitasi dan proses dapat berjalan lebih efektif.

Kata kunci : vehicle reservation systems, reservasi, PT.Pertamina EP

**THE DEVELOPMENT OF VEHICLE
RESERVATION SYSTEMS USING BUSINESS
PROCESS MANAGEMENT APPROACH
(CASE STUDY : PT.PERTAMINA EP)**

Name : JOSHUA SATRIAWAN
NRP : 5212 100 137
Departement : INFORMATION SYSTEM FTIF-ITS
Supervisor 1 : Arif Wibisono, S.Kom., M.Sc.

ABSTRACT

Vehicle reservation system is very important for any company to work so that employees can use a company vehicle for business purposes office property offices. However, there are some constraints in the system, namely the number of vehicles and employees are very much rely on this vehicle reservation system, moreover, is not there a software or applications that support this system.

This thesis discusses the development of the vehicle reservation system (vehicle reservation system) using a process-oriented approach. In this application the system will work online and connect between the user (the person who make the request) and approvers (people who respond to the request / provide feedback). The reason why this study discusses using process-oriented method for the application of vehicle reservation system is an application to the approval process. Further discipline that is used is the business process management (BPM), these disciplines are used to develop this application because the process instance contained in the BPM is very clearly described and suitable to be applied in this case study.

The results in this study is a web-based application vehicle reservation system developed using Bonitasoft, one of the recognized BPM software in the world, using a case study PT.Pertamina EP which is one of the largest oil company in Indonesia. This application can connect all actors involved in the business process, so that the problem regarding the reservation lending vehicle will teratasi.Dengan this application then processes running may ter-automation so that actors who undertake this process be facilitated and the process can be run more effectively.

Keywords: *vehicle reservation systems, reservation, PT.Pertamina EP*

DAFTAR ISI

LEMBAR PENGESAHAN.....	iii
LEMBAR PERSETUJUAN.....	iv
ABSTRAK	v
ABSTRACT.....	vii
KATA PENGANTAR	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah	3
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian	4
1.5. Manfaat Penelitian	4
1.6. Relevansi.....	5
BAB II TINJAUAN PUSTAKA.....	7
2.1. Studi Sebelumnya	7
2.2. Dasar Teori	8
2.2.1. Business Process Management	8
2.2.2. Bonita BPM.....	15
2.2.3. <i>Vehicle Reservation Systems</i>	16
BAB III METODOLOGI PENELITIAN.....	23
3.1. Studi Literatur	23
3.2. <i>Process Identification</i>	24
3.3. <i>Process Discovery</i>	24
3.4. <i>Process Implementation</i>	24
3.5. Penyusunan Dokumen Tugas Akhir	25
BAB IV PERANCANGAN	27
4.1. <i>Process Discovery</i>	27
4.2. Business Process Model Notation.....	29
4.2.1 Penentuan <i>Actor</i>	29
4.2.2 Pembuatan Proses Model	31
4.3. <i>Generate Bonita Software</i>	40
4.3.1. Deklarasi Organisasi	40

4.3.2.	<i>Publish Organization</i>	43
4.3.3.	<i>Configure</i>	46
BAB V IMPLEMENTASI		49
5.1	Membuat <i>Form</i>	49
5.1.1.	Memasukkan <i>available values</i>	52
5.1.2.	Menggunakan <i>decision XOR</i>	56
5.1.3.	Membuat <i>Terminate and End</i> dan <i>End</i>	59
BAB VI HASIL DAN PEMBAHASAN		63
6.1	Permodelan <i>as-is</i> Sistem Reservasi Kendaraan PT.PERTAMINA EP	63
6.2	Automasi Sistem Reservasi (<i>as-is</i>) PT.PERTAMINA EP	63
BAB VII KESIMPULAN DAN SARAN		73
7.1	Kesimpulan	73
7.2	Saran	74
7.2.1.	<i>Leverage</i>	74
7.2.2.	<i>Escalate</i>	74
DAFTAR PUSTAKA.....		75
LAMPIRAN A		A-1
1.1	Process Analysis.....	A-1
1.2	Process Re-Design.....	A-3
1.3	Pembahasan Proses Model <i>To-Be</i>	A-7
LAMPIRAN B.....		B-1
2.1	Pertanyaan Pengantar	B-1
2.2	Pertanyaan Teknis	B-1
LAMPIRAN C.....		C-1
3.1	Terdapat Kendaraan Yang Tersedia	C-1
3.2	Tidak Terdapat Kendaraan Yang Tersedia.....	C-5
BIODATA PENULIS		77

DAFTAR GAMBAR

Gambar 1.1 Riset Laboratorium Sistem Enterprise.....	5
Gambar 2.1 BPM <i>lifecycle</i>	14
Gambar 2.2 Struktur organisasi Departemen SCM.....	20
Gambar 3.1 Metodologi Tugas Akhir	23
Gambar 4.1 Admin Supir Pertamina EP	27
Gambar 4.2 Kondisi Basement Pertamina	28
Gambar 4.3 Actor untuk role staff	30
Gambar 4.4 Actor untuk role manager.....	30
Gambar 4.5 Actor untuk role VP	30
Gambar 4.6 Variable yang digunakan	32
Gambar 4.7 Proses Model as-is (role : staff).....	36
Gambar 4.8 Proses Model as-is (role : manager).....	37
Gambar 4.9 Proses Model as-is (role : VP).....	38
Gambar 4.10 Deklarasi Organisasi (langkah pertama)	40
Gambar 4.11 Deklarasi Organisasi (langkah kedua).....	41
Gambar 4.12 Deklarasi Organisasi (langkah ketiga).....	42
Gambar 4.13 Deklarasi Organisasi (langkah keempat).....	43
Gambar 4.14 Publish Organisasi (langkah pertama).....	44
Gambar 4.15 Publish Organisasi (langkah kedua)	45
Gambar 4.16 Publish Organisasi (langkah ketiga).....	45
Gambar 4.17 <i>Configure</i> (actor mapping).....	46
Gambar 5.1. Proses Model <i>as-is</i>	50
Gambar 5.2. <i>Add form</i> (ceklist <i>variable</i>) pada <i>case</i> "mengisi <i>form</i> "	51
Gambar 5.3. <i>Edit Form</i> pada <i>case</i> "mengisi <i>form</i> "	52
Gambar 5.4. <i>Properti</i> dari <i>Variable</i> "jenis_keberangkatan" ..	53
Gambar 5.5 <i>Property</i> dari <i>Variable</i> "pilihan_jenis_keberangkatan".....	53
Gambar 5.7. Pilihan dari <i>variable</i> "pilihan_jenis_kendaraan"	54
Gambar 5.8. <i>Property</i> dari <i>Variable</i> "jenis_keberangkatan" pada halaman <i>edit form</i>	55
Gambar 5.9. Pilihan untuk mengisi nilai <i>multiple</i> pada <i>variable</i> "jenis_keberangkatan"	55

Gambar 5.10. <i>Decision XOR</i>	56
Gambar 5.11. Add form (ceklist variable) pada case "menanggapi form"	57
Gambar 5.12. Halaman <i>edit form</i> pada case "menanggapi form"	57
Gambar 5.13. <i>Property</i> pada garis "YES"	58
Gambar 5.14. <i>Property</i> pada garis "NO"	59
Gambar 5.15. Proses Model <i>as-is</i>	61
Gambar 6.1. Hasil Permodelan <i>as-is</i> Sistem Reservasi Kendaraan PT.PERTAMINA EP	64
Gambar 6.2. Tampilan dari case "mengisi form"	65
Gambar 6.3. Tampilan dari case "menanggapi form"	66
Gambar 6.4. Tampilan dari case "mengisi voucher taksi"	66
Gambar 6.5. Tampilan dari case "menerima kode voucher taksi"	67
Gambar 6.6. Tampilan dari case "mengisi data supir"	67
Gambar 6.7. Tampilan dari case "membaca <i>feedback</i> "	68
Gambar 6.8. Tampilan dari case "melakukan <i>check-in</i> kendaraan"	68
Gambar 6.9. Tampilan dari case "mengkonfirmasi <i>check-in</i> "	69
Gambar 6.10. Tampilan dari case "update status kendaraan <i>check-in</i> "	69
Gambar 6.11. Tampilan dari case "melakukan <i>check-out</i> kendaraan"	70
Gambar 6.12. Tampilan dari case "Mengkonfirmasi <i>check-out</i> "	70
Gambar 6.13. Tampilan dari case "update status kendaraan <i>check-out</i> "	71
Gambar A.1. Proses model <i>to-be</i>	10A-1
Gambar C.2. case "mengisi form"	C-2
Gambar C.3. case "menanggapi form"	C-2
Gambar C.4. case "mengisi data supir"	C-2
Gambar C.5. case " <i>Pre Check-in</i> "	C-3
Gambar C.6. case "melakukan <i>check-in</i> kendaraan"	C-3
Gambar C.7. case "mengkonfirmasi <i>check-in</i> "	C-3
Gambar C.8. case " <i>update</i> status kendaraan (<i>check-in</i>)"	C-4
Gambar C.9. case "melakukan <i>check-out</i> kendaraan"	C-4

Gambar C.10. <i>case</i> "mengkonfirmasi <i>check-out</i> "	C-4
Gambar C.11. <i>case</i> " <i>update</i> status kendaraan (<i>check-out</i>)"	C-4
Gambar C.12. <i>case</i> "mengisi <i>form</i> "	C-5
Gambar C.13. <i>case</i> "menanggapi <i>form</i> "	C-6
Gambar C.14. <i>case</i> "mengisi kode <i>voucher</i> taksi"	C-6
Gambar C.15. <i>case</i> "menerima kode <i>voucher</i> taksi"	C-6

DAFTAR TABEL

Table A.1 <i>Value-added table</i>	A-2
Table A.2. <i>7 heuristics table</i>	A-5
Table A.3. penerapan <i>heuristics</i> pada proses model <i>to-be</i> ..	A-7
Table C.1 tabel <i>case</i> dan <i>actor (testing 1)</i>	C-1
Table C.2 tabel <i>case</i> dan <i>actor (testing 2)</i>	C-5

BAB I

PENDAHULUAN

Pada bagian ini akan dijelaskan mengenai latar belakang masalah, rumusan masalah, batasan masalah dan tujuan penelitian yang mendasari penelitian tugas akhir.

1.1. Latar Belakang

PT PERTAMINA EP adalah perusahaan yang menyelenggarakan kegiatan usaha di sektor hulu bidang minyak dan gas bumi, meliputi eksplorasi dan eksploitasi. Di samping itu, PT.PERTAMINA EP juga melaksanakan kegiatan usaha penunjang lain yang secara langsung maupun tidak langsung mendukung bidang kegiatan usaha utama [1]

Wilayah kerja Perusahaan saat ini terbagi ke dalam 5 Aset – yang mencakup Sumatera, Jawa dan Kawasan Timur Indonesia. Kegiatan operasi kelima Aset tersebut adalah [1] :

1. Aset 1 yang mengelola wilayah kerja mulai dari Sumatera Bagian Utara sampai dengan Sumatera Bagian Selatan dan terbagi dalam sejumlah area operasi meliputi Lapangan Rantau, Lapangan Pangkalan Susu, Lapangan Lirik, Lapangan Jambi dan Lapangan Ramba.
2. Aset 2 yang mengelola wilayah kerja di Sumatera Selatan dan terbagi dalam sejumlah area operasi meliputi Lapangan Prabumulih, Lapangan Pendopo, Lapangan Limau dan Lapangan Adera.
3. Aset 3 yang mengelola wilayah kerja di Jawa Barat dan terbagi dalam sejumlah area operasi meliputi Lapangan Subang, Lapangan Jatibarang, Lapangan Tambun.
4. Aset 4 yang mengelola wilayah kerja di Jawa Tengah dan Jawa Timur yaitu Lapangan Cepu.

5. Aset 5 yang mengelola wilayah kerja di Kawasan Timur Indonesia dan terbagi dalam sejumlah area operasi meliputi Lapangan Sangatta, Lapangan Sangasanga, Lapangan Tanjung, Lapangan Tarakan, Lapangan Bunyu & Lapangan Papua [1].

Dapat dilihat bahwa PT. PERTAMINA EP mempunyai banyak sekali kantor cabang yang tersebar di berbagai daerah (kantor pusat berada di DKI Jakarta). Hal ini juga tidak terlepas dari jumlah karyawan yang bekerja pada PT. PERTAMINA EP seluruh kantor cabang dan kantor pusat. Dengan melihat 2 hal tersebut maka bisa disimpulkan pula bahwa urusan PT. PERTAMINA EP dengan stakeholder eksternal juga cukup banyak, baik itu di dalam kantor maupun luar kantor. Banyak juga fasilitas antar-jemput yang disediakan oleh PT. PERTAMINA EP yang dapat digunakan oleh para karyawannya dengan tujuan agar karyawan tersebut tidak terlalu kelelahan dengan pekerjaan kantornya.

Dengan kondisi tersebut tentunya PT. PERTAMINA EP memiliki banyak transportasi untuk memfasilitasi karyawannya agar dapat bekerja lebih baik lagi. Transportasi-transportasi ini dapat berbagai macam bentuknya seperti mobil, motor, ataupun kendaraan berat lainnya. Namun tidak semua transportasi PT. PERTAMINA EP merupakan hak milik perusahaan, terdapat beberapa transportasi yang merupakan hasil peminjaman dari kontrak dengan beberapa vendor, sehingga dapat dilihat bahwa transportasi milik PT. PERTAMINA EP sangatlah banyak.

Sebuah masalah muncul ketika seorang karyawan ingin melakukan peminjaman transportasi untuk keperluan kantor (contoh : rapat dengan *stakeholder* yang merupakan urusan kantor, rapat diadakan di luar kantor sehingga membutuhkan kendaraan), masalah tersebut adalah pada PT. PERTAMINA EP belum terdapat sistem bagi karyawan untuk melakukan peminjaman transportasi secara online ataupun otomatis, proses peminjaman kendaraan ini masih dilakukan secara

manual sehingga sangat merepotkan dan pendataan-pun tidak dilakukan dengan baik. Selain itu jika dilihat dari sisi supir ataupun mandor (yang merupakan atasan dari supir) hal ini juga tidak baik karena mereka tidak dapat mengetahui secara cepat jika terdapat karyawan yang membutuhkan peminjaman transportasi. Proses yang selama ini terjadi di PT. PERTAMINA EP ini sangatlah rumit untuk melakukan sebuah reservasi peminjaman transportasi, seharusnya ini tidak menjadi sebuah masalah karena permasalahan seperti ini dapat diselesaikan dengan menggunakan IT.

1.2. Perumusan Masalah

Berdasarkan penjelasan latar belakang di atas, rumusan masalah yang menjadi fokus utama dalam tugas akhir ini adalah:

1. Bagaimana memodelkan sistem reservasi kendaraan (*as-is*) di PT.PERTAMINA EP ?
2. Bagaimana mengautomasi sistem reservasi (*as-is*) di PT.PERTAMINA EP ?

1.3. Batasan Masalah

Dari permasalahan yang disebutkan di atas, batasan masalah dalam tugas akhir ini adalah:

1. Aplikasi yang dikembangkan hanya ditujukan kepada PT.PERTAMINA EP.
2. Aplikasi dikembangkan menggunakan *BonitaSoft*
3. Pada penelitian ini tidak dilakukan tahapan *process monitoring and controlling*.
4. Pada penelitian ini yang dibahas adalah proses model *as-is*, proses model *as-is* ini akan dimodelkan dan diimplementasikan.

5. Pengembangan aplikasi ini difokuskan terhadap reservasi peminjaman kendaraan kantor PT. PERTAMINA EP.

1.4. Tujuan Penelitian

Tujuan dari pengerjaan tugas akhir ini adalah :

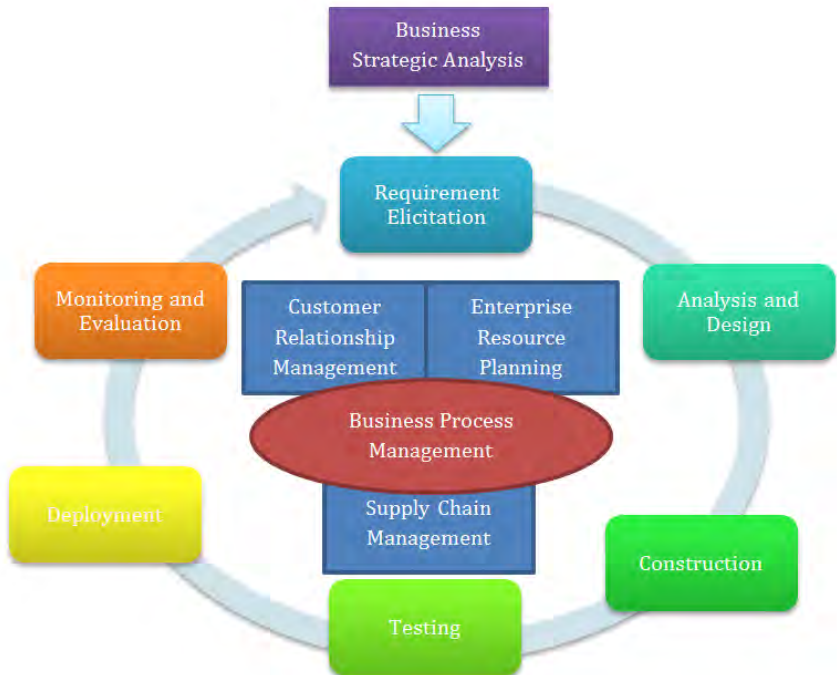
1. Memodelkan proses reservasi kendaraan (*as-is*) di PT.PERTAMINA EP.
2. Melakukan tahap implementasi terhadap proses reservasi kendaraan (*as-is*) di PT.PERTAMINA EP.
3. Memudahkan (mengautomasi) karyawan PT.PERTAMINA EP untuk melakukan reservasi kendaraan di perusahaan.

1.5. Manfaat Penelitian

Melalui tugas akhir ini diharapkan dapat memberi manfaat yaitu:

1. Seluruh karyawan PT.PERTAMINA EP Indonesia dapat melakukan reservasi peminjaman transportasi PT. PERTAMINA EP (yang ada pada kantornya masing-masing).
2. Dapat mengetahui status kendaraan-kendaraan yang dimiliki oleh PT. PERTAMINA EP.

1.6. Relevansi



Gambar 1.1 Riset Laboratorium Sistem Enterprise

Topik tugas akhir yang dikerjakan oleh penulis adalah mengenai *Business Process Management* yang merupakan salah satu riset pada laboratorium Sistem Enterprise. Pada laboratorium Sistem Enterprise ini terdapat 4 pokok topic yaitu *customer relationship management (CRM)*, *enterprise resource planning (ERP)*, *supply chain management (SCM)* dan *business process management (BPM)*.

Seperti yang bisa dilihat pada gambar diatas bahwa terdapat *lifecycle* dari kerangka kerja riset laboratorium Sistem Enterprise dengan 4 topik besar yaitu CRM, ERP, SCM dan BPM. Pada laboratorium ini ditekankan mengenai visibilitas dari topik tugas akhir yang diajukan oleh mahasiswanya, sehingga setelah mahasiswa yang bersangkutan tersebut lulus atau meninggalkan kehidupan kampus maka pihak dosen

ataupun laboratorium dapat melanjutkan riset yang sudah dilakukan oleh mahasiswa tersebut, karena pastinya riset tersebut dapat dilanjutkan berdasarkan *lifecycle* dari kerangka kerja riset laboratorium sistem enterprise ini.

Topik BPM sendiri dapat menggunakan *full cycle* dari kerangka kerja riset tersebut, namun dapat juga dilakukan secara tidak *full cycle* dari kerangka tersebut karena riset tersebut masih bisa dilanjutkan oleh mahasiswa yang ingin melakukan riset pada laboratorium sistem enterprise selanjutnya.

BAB II

TINJAUAN PUSTAKA

Pada bab ini dijelaskan mengenai teori-teori terkait yang bersumber dari buku, jurnal, ataupun artikel yang berfungsi sebagai dasar dalam melakukan pengerjaan tugas akhir agar dapat memahami konsep atau teori penyelesaian permasalahan yang ada.

2.1. Studi Sebelumnya

Penelitian-penelitian yang memiliki topik yang sama dengan penelitian ini (*vehicle reservation system*) yang dilakukan sebelum penelitian ini sudah sangat banyak, hasil dari penelitian tersebut berupa sebuah aplikasi untuk reservasi mobil pada sebuah agen travel ataupun sebuah aplikasi yang berfungsi untuk melakukan pemesanan paket wisata secara online.

Sebuah penelitian dilakukan menggunakan studi kasus “sabila transport” yang merupakan salah satu penyedia jasa penyewaan kendaraan bermotor dan penyedia jasa paket wisata yang belum memiliki fasilitas pemesanan untuk pelanggan tanpa datang langsung ke kantor. Pelanggan harus merelakan waktunya untuk datang ke kantor Sabila Transport untuk memesan kendaraan maupun paket wisata yang diinginkan. Untuk itu dibutuhkan aplikasi berbasis web berupa website e-commerce yang mampu memberikan informasi yang lengkap dan akurat kepada pelanggan melalui Internet kapanpun dan dimanapun pelanggan berada. Dengan adanya website *e-commerce* pelanggan bisa melakukan pemesanan secara online tanpa harus mendatangi tempat penyedia jasa itu berada. Selain melakukan pembelian dan pemesanan secara online, fasilitas yang dapat dimiliki sebuah website e-commerce adalah adanya transaksi pembayaran dengan cara online maupun off-line melalui transfer antar rekening bank. Dari penelitian diatas ini dapat dilihat bahwa produknya adalah sebuah aplikasi yang melakukan sebuah pemesanan

terhadap kendaraan ataupun jasa paket wisata, tujuan dari produk ini adalah mendukung bisnis utama dari studi kasus yang diambil. Sistemasi yang digunakan pada aplikasi ini adalah sebuah request-approval yang dilakukan oleh customer dan juga pemilik usaha yaitu “sabila transport” [2]

Penelitian lainnya yang serupa adalah pengembangan sebuah aplikasi reservasi sewa mobil online yang menggunakan studi kasus “rental barokah” berbasis. Aplikasi ini hampir sama dengan aplikasi yang sudah dibahas diatas, aplikasi ini memungkinkan para customer untuk melakukan reservasi mobil secara online melalui sebuah website. Sistemasi dari website ini juga sama yaitu request approval yang melibatkan antara customer dengan pihak “rental barokah” ini. Pengembangan aplikasi ini menggunakan coding melalui MYSQL, XAMPP, APACHE, PHP, HTML, CSS [3]

Selain itu terdapat juga penelitian lainnya yang serupa dan menggunakan tools pengembangan menggunakan virtual box. Walaupun sebenarnya penelitian ini hanya sebagai tugas kuliah biasa tanpa menggunakan studi kasus namun pengembangan aplikasi ini sudah banyak dilakukan dan beberapa diantaranya ada yang menggunakan tools virtual box. Tentunya aplikasi ini juga memperbolehkan para customer untuk dapat melakukan pemesanan kendaraan secara online..

2.2. Dasar Teori

2.2.1. Business Process Management

Business Process Management (BPM) merupakan sebuah seni dan ilmu yang melihat bagaimana sebuah pekerjaan dilakukan pada sebuah organisasi untuk memastikan hasil yang konsisten dan mengambil keuntungan dari sebuah perbaikan / perubahan / inovasi [4]. BPM merupakan suatu metode penyelarasan secara efisien suatu organisasi dengan keinginan dan kebutuhan organisasi tersebut. BPM merupakan suatu

pendekatan manajemen holistic untuk meningkatkan efektivitas dan efisiensi bisnis seiring upaya untuk mencapai inovasi, fleksibilitas dan integrasi dengan teknologi. BPM berupaya untuk melakukan perbaikan proses secara berkelanjutan atau bisa juga disebut sebagai suatu proses “optimalisasi proses”.

Business Process Management Systems (BPMS) adalah sebuah *software* yang terintegrasi yang dapat mengkoordinasikan dan mengatur sebuah pekerjaan melalui beberapa aktifitas yang sudah ditetapkan [5]. Terdapat juga sumber yang mengatakan serupa bahwa BPM adalah perangkat lunak yang didorong oleh proses representasi eksplisit untuk mengkoordinir keberlakuan / aplikasi dari proses bisnis [6]. Sumber lainnya mengatakan bahwa BPM adalah satu set metode, alat, dan teknologi yang digunakan untuk merancang, membuat, menganalisis, dan mengendalikan proses bisnis operasional. BPM adalah proses pendekatan sentris untuk meningkatkan kinerja yang menggabungkan teknologi informasi dengan proses dan tata kelola metodologi. BPM adalah kolaborasi antara orang-orang bisnis dan teknologi informasi untuk mendorong efektif, tangkas, dan transparan proses bisnis. BPM meliputi orang, sistem, fungsi, bisnis, pelanggan, pemasok, dan mitra [7].

Selain itu terdapat juga sumber yang mengatakan bahwa BPM memiliki karakteristik sumber daya yang dapat dilihat dari dua sudut pandang yaitu sudut pandang sumber daya dan sudut pandang orang yang bertanggung jawab atas proses tersebut [8].

BPM sebenarnya berawal dari total quality management and process business re-engineering. BPM adalah IT yang dimotori oleh disiplin manajemen yang mempromosikan kelincihan organisasi dengan secepat terangnya dan mendukung upaya masyarakat untuk mendorong perubahan proses dan inovasi yang cepat. BPM merupakan pendekatan manajemen yang

berfokus pada segala aspek organisasi yang melibatkan apa yang customer inginkan dan customer butuhkan. BPM menggabungkan pendekatan proses-sentris dan lintas fungsional untuk meningkatkan cara organisasi mencapai tujuan mereka. Sehingga dapat diringkaskan bahwa BPM adalah satu kesatuan yang mencakup : Manajemen disiplin, metodologi untuk mengelola proses bisnis dan teknologi atau set (kumpulan) teknologi.

Bisnis Proses (bukan BPM) mungkin sesuatu yang sudah umum dalam perusahaan, tujuan dari bisnis proses adalah untuk meningkatkan efektifitas dan efisiensi dalam perusahaan, dan tentunya berujung pada meningkatnya persaingan terhadap kompetitor dan berakhir pada meningkatnya bisnis sebuah perusahaan.

Seiring berkembangnya persaingan dalam dunia bisnis, setiap perusahaan harus cepat dalam merespon, baik terhadap kompetitor maupun pelanggan. Hal ini tentunya akan berimbas kepada bisnis proses yang akan semakin kompleks dan memakan banyak resources. Padahal untuk membuat desain bisnis proses saja bisa memakan waktu berbulan-bulan. Hal ini sangat tidak efisien, sementara disisi lain pelanggan tidak bisa menunggu. Berikut ini adalah permasalahan yang sering dihadapi dalam membuat business process tanpa BPM :

- a. Banyak duplikasi data : banyak input data yang berulang
- b. Inefficient : banyak pekerjaan yang berulang dan orang yang mengerjakan hal yang sama
- c. Manajemen sulit mengontrol : aliran tugas sulit dikontrol oleh manajemen
- d. Tidak bisa melihat proses yang terjadi : proses yang terjadi di tiap departemen sulit terlihat
- e. Unstructured task, kinerja hanya lewat mulut kemulut tanpa ada data : pekerjaan tidak terdata

- f. Visibillity terbatas : sulit menentukan SLA sebuah pekerjaan, misal dalam sebuah bank ketika menangani complain kartu ATM rusak
- g. Kesulitan administrative

Kini di dunia IT sudah dikenal istilah Business Process Management (BPM), tentunya dengan adanya BPM bisa meminimalisir bahkan menghilangkan masalah-masalah diatas. Yang BPM tidak sama dengan tools bisnis proses yang ada sebelumnya (contoh : ms.visio). BPM dapat meng-cover semua pekerjaan dalam bisnis rposes, berbeda dengan business process tools yang hanya bisa dipakai pada saat Process Modeller.

Dalam pencapaian hasil yang diharapkan oleh perusahaan-perusahaan yakni kesuksesan dalam bisnisnya dan pertahanan yang kuat untuk tetap berdiri di tengah-tengah persaingan atas pasar dan pelanggan disertai dengan kekuatan atas produk atau jasa yang dikenalkan oleh perusahaan tersebut, sangat dibutuhkan yang namanya penciptaan, pengontrolan, perubahan dan perbaikan atas bentuk strategi bisnis yang dijalankan sehingga menjadi lebih mudah dan fleksibe. Dikarenakan masih banyaknya perusahaan-perushaan yang masih menggunakan otomasi secara manual dan sebagian perusahaan lainnya masih menggunakan otomasi secara parsial. BPM diharapkan dapat mengatasi masalah tersebut, sebagai jalan untuk menciptakan suatu kondisi dimana otomasi suatu bisnis dapat dilakukan secara cepat terhadap perubahan kondisi bisnis yang akan terwujud. BPM sendiri mendukung yang namanya lifecycle dan bersifat loop tertutup yang memungkinkan pembangunan sistem yang elbih efisien dan efektif. Bentuk kegiatan perubahan, penciptaan, pengontrolan, perubahan dan perbaikan adalah proses bisnis yang dijalankan oleh perusahaan. Proses bisnis merupakan kunci utama yang diperlukan untuk menjalankan suatu perusahaan dalam pencapaiannya untuk menuju kesuksesan dan bertahan di tengah persaingan pasar. Proses bisnis tersebut tidaklah

dijalankan secara brutal atau secara berantakan. Karena itulah diperlukannya suatu manajemen di dalamnya.

Namun kenyataannya tidak semua perusahaan yang mengerti dan mampu untuk mengontrol proses bisnis yang dimilikinya. Mungkin pihak manajemen telah berhasil untuk membuat dan membentuk proses bisnis yang ideal yang dapat dijalankan, tetapi kenyataan yang terjadi di lapangan tidak seperti yang diharapkan dan sesuai dengan yang telah dibentuk oleh pihak manajemen. Pengelolaan proses bisnis yang baik akan menciptakan suatu peluang bagi perusahaan, dalam pelaksanaan proses bisnis kadang terjadi stagnasi, redundansi, ke-tidak-efisienan, dan kesalahan-kesalahan lainnya yang tidak dapat diantisipasi sebelumnya. Kegiatan bisnis yang dimana contohnya ialah pengadaan produk, pengadaan supply, perekrutan karyawan dan lain-lainnya, jika tidak dijalankan sesuai dengan manajemen yang baik dan yang telah dibentuk agar kegiatannya berjalan dengan efisien dan efektif akan menjadi penghalang bagi perusahaan untuk mencapai tujuan inti perusahaan tersebut.

Berikut ini adalah keuntungan yang akan didapatkan ketika perusahaan menerapkan BPM :

- a. Solusi BPM akan memfasilitasi perusahaan dalam memodelkan proses bisnis yang dimiliki, mengotomatisasi jalannya proses bisnis tersebut, memonitor jalannya proses, serta memberikan cara yang mudah dan cepat ketika perusahaan akan melakukan perubahan proses bisnis untuk meningkatkan performansinya
- b. Software BPM membantu perusahaan untuk mengotomatisasi tugas-tugas yang selama ini masih dilakukan secara manual. Solusi BPM dapat mengotomatisasi proses persetujuan serta penolakan, notifikasi dan laporan status
- c. Dengan BPM, integrasi antar proses bisnis dapat dilakukan dengan mudah dan cepat

- d. BPM membantu perusahaan dalam membuat exception handling dan proses alternatif yang sangat dibutuhkan untuk menanggulangi masalah pada bisnis yang bersifat sangat dinamis seperti sekarang ini
- e. BPM dapat meningkatkan daya respon bisnis melalui kemampuan untuk mendapatkan informasi dengan cepat dan real-time.
- f. BPM mengurangi waktu yang dibutuhkan pada pelaksanaan suatu proses bisnis.
- g. BPM meningkatkan produktivitas setiap karyawan
- h. Umumnya proses bisnis tentunya membutuhkan banyak orang dan sumber daya. Sebuah solusi BPM yang baik dapat mengurangi jumlah sumber daya yang dibutuhkan pada sebuah proses
- i. Didalam melakukan manajemen proses bisnis, ada proses lifecycle yang terjadi, seperti definisi proses, eksekusi dan control proses, hingga monitoring dan perbaikan proses. Full process lifecycle ini akan dapat membantu perusahaan dalam peningkatan kualitas bisnis yang saling berkesinambungan.

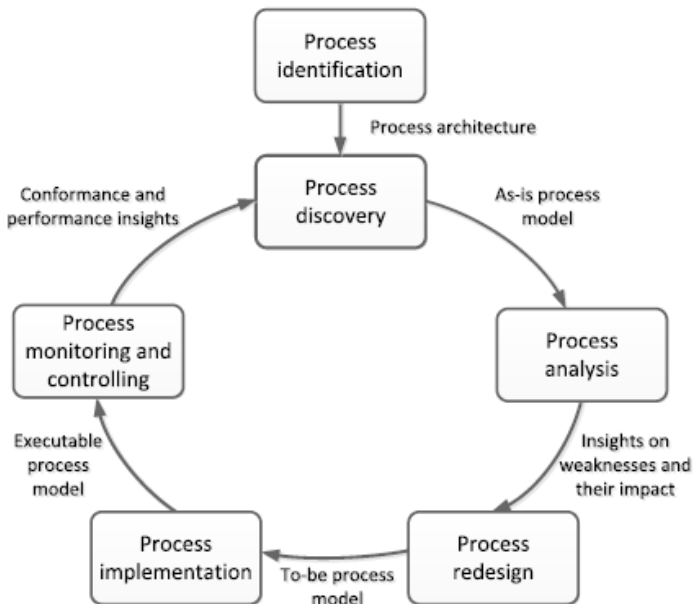
a. *Process identification*

Fase ini fase pertama, dimana sebuah masalah bisnis akan diangkat / diajukan untuk dianalisis. Proses-proses yang ada akan diidentifikasi, dipisahkan dan dihubungkan satu dengan yang lainnya. Hasil dari fase ini adalah sebuah arsitektur yang menggambarkan keseluruhan proses yang ada dengan organisasi dan hubungannya (process landscape).

b. *Process discovery*

Fase ini melanjutkan hasil dari fase sebelumnya (process identification), pada fase ini akan dilakukan semacam survey atau koreksi yang dapat menyempurnakan process landscape yang sudah dibuat sebelumnya

Selanjutnya akan ditampilkan gambar 2.1 yaitu *lifecycle* dari BPM dan penjelasan mengenai fase-fase dari BPM [4]:



Gambar 2.1 BPM lifecycle

c. *Process analysis*

Fase ini merupakan fase pembuatan business process model and notation (BPMN) dalam bentuk as-is

d. *Process re-design*

Fase ini adalah fase yang menganalisa BPMN bentuk as-is yang sudah dibuat sebelumnya, analisa yang dilakukan adalah analisis apakah proses yang ada sudah efektif dan efisien, setelah itu pada fase ini juga akan dibentuk BPMN dalam bentuk to-be.

Terdapat total 7 kategori *heuristics process redesign* yang dapat digunakan sebagai acuan untuk melakukan *process redesign* ini sendiri yaitu:

- *Customer heuristics*

- *Business process operation heuristics*
- *Business process behavior heuristics*
- *Organization heuristics*
- *Information heuristics*
- *Technology heuristics*
- *External environment heuristics*

Setiap kategori tersebut memiliki sub-kategori lagi dengan total 29 sub-kategori pada 7 kategori *heuristics process redesign* tersebut.

e. *Process implementation*

Fase ini adalah fase implementasi dari bentuk BPMN yang sudah dibentuk pada tahap-tahap sebelumnya. Pada studi kasus tugas akhir kali ini, penulis akan menggunakan software yang bernama Bonitasoft sehingga fase implementasi ini akan menggunakan bonitasoft.

f. *Process monitoring and controlling*

Fase ini adalah fase pengawasan, fase ini dilakukan setelah implementasi sudah selesai dilakukan dan sebuah produk sudah dihasilkan sehingga produk tersebut dapat digunakan dan dilihat penerapan dalam realitanya. Jika memang terdapat sesuatu yang kurang maka dapat dilakukan kembali process identification [4].

2.2.2. Bonita BPM

Bonita BPM merupakan sebuah open-source BPM dan workflow yang dibuat pada tahun 2001. Pada awalnya Bonita BPM ini ditemukan oleh France National Institute for Research in Computer Science, dan kemudian di-inkubator selama beberapa tahun di dalam French Computer Science Company. Sejak tahun 2009 pengembangan Bonita ini didukung oleh sebuah perusahaan yang tertarik pada aktivitas ini yaitu *Bonitasoft*. Bonita BPM mempunyai 3 komponen, yaitu [9]:

a. *Bonita Studio*

Memungkinkan user untuk memodifikasi atau mengubah business processes secara grafik dengan mengikuti BPMN standart. User juga dapat mengkoneksikan proses yang ada dengan komponen information system yang lainnya (contoh: messaging, enterprise resource planning, enterprise content management, dan database) dengan tujuan untuk mengenerate aplikasi dan bisa diakses sebagai bentuk web.

b. *Bonita BPM engine*

Merupakan sebuah aplikasi JAVA yang mengeksekusi definisi proses yang dibuat dengan Bonita Studio.

c. *Bonita portal*

Merupakan portal yang memungkinkan setiap end-user untuk mengelola dalam webmail-like interface untuk semua tugas yang ia (end-user)punyai. Portal ini juga memungkinkan pemilik proses untuk mengelola dan mendapatkan laporan tentang proses yang bersangkutan.

Sumber lain mengatakan bahwa *Bonita Open Source* (BOS) merupakan solusi open source untuk proses otomatisasi dan *support*. Tidak seperti BPMS komersial lainnya, BOS memiliki beberapa kelebihan dan kekurangan yang khas untuk solusi perangkat lunak yang lebih kecil berdasarkan teknologi open source [10].

2.2.3. *Vehicle Reservation Systems*

Aplikasi *Vehicle Reservation systems* ini merupakan sebuah aplikasi berbasis Web yang dirancang oleh PT.PERTAMINA EP yang berfungsi untuk meng-integrasikan data-data transportasi yang ada di setiap kantor cabang Pertamina EP di Indonesia sehingga mempermudah proses reservasi transportasi.

2.2.3.1. *Process Identification*

Process Identification merupakan tahapan pertama untuk memulai analisis proses model menggunakan pendekatan *Business Process Management* (BPM). Analisis pertama yang akan dibahas adalah menentukan *process instance*, *process instance* merupakan sebuah objek / subjek yang menjadi produk / jasa yang ditawarkan pada sebuah proses bisnis, *process instance* juga merupakan suatu hal (dapat berupa objek ataupun subjek) yang bergerak dari satu *case* menuju *case* lainnya dari proses bisnis suatu organisasi / perusahaan.

Pada PT.PERTAMINA EP khususnya pada proses reservasi kendaraan, *process instance* dapat ditentukan melalui analisis proses bisnis yang sedang terjadi. Setelah dilakukan survey maka diketahui bahwa proses reservasi kendaraan di PT.PERTAMINA EP adalah sebagai berikut :

1. Karyawan melakukan reservasi (reservasi dapat dilakukan melalui telfon / *byphone* ataupun dengan mengirimkan email) dengan mencantumkan ID karyawan, destinasi, jam keberangkatan, jenis reservasi (antar-jemput atau sekali jalan), email atau telfon ditujukan kepada admin kendaraan (admin supir) yang merupakan karyawan dari departemen *Supply Chain Management* (SCM).
2. Admin kendaraan dari departemen SCM menerima *request* dari karyawan yang bersangkutan, setelah itu melihat apakah terdapat kendaraan yang kosong berdasarkan jabatan dari karyawan tersebut (jabatan karyawan dikelompokkan menjadi 3 yaitu, staff, manager, dan VP atau GM) karena setiap jabatan mendapatkan jatah kendaraan yang berbeda.
3. Terdapat dua kemungkinan *case* yang terjadi setelah in, yang pertama jika terdapat kendaraan yang dapat disediakan untuk karyawan yang bersangkutan (ada atau tidaknya berdasarkan status karyawan), yang kedua jika tidak terdapat kendaraan yang tersedia

untuk memenuhi *request* karyawan tersebut (berdasarkan jabatannya) maka karyawan yang bersangkutan akan diberikan *voucher* taksi sehingga karyawan tersebut memperoleh fasilitas untuk menaiki taksi sehingga dapat menggantikan kendaraan kantor yang sedang tidak tersedia (jika yang terjadi adalah *case* kedua ini maka proses berakhir disini)

4. Jika yang terjadi adalah *case* pertama maka departemen SCM akan memberikan *feedback* kepada karyawan tersebut dan mengabarkan bahwa *request* yang dilakukan sudah diterima dan disetujui. Setelah itu karyawan akan menggunakan fasilitas kendaraan tersebut pada jam yang telah ditentukan
5. Bergantung terhadap jenis reservasi yang diminta oleh karyawan, jika jenis yang diminta adalah sekali jalan, maka supir hanya bertugas untuk mengantarkan karyawan ke tempat tujuan dan supir dapat kembali ke kantor, namun jika jenis reservasi yang diminta adalah antar-jemput maka supir bertugas untuk mengantarkan karyawan ke tempat tujuan dan supir jugawajib untuk menjemput karyawan pada jam yang sudah dijanjikan.
6. Setelah itu ketika supir sudah kembali maka supir dan karyawan yang menggunakan fasilitas kendaraan kantor tersebut harus melaporkan kepada admin supir bahwa kendaraan sudah kembali (karyawan memberikan *feedback* atau laporan kepada admin supir bahwa pemakaian sudah selesai).

Selanjutnya adalah menentukan *case type* dan *business function*, untuk melakukan hal ini maka diperlukan beberapa analisis juga, berikut pembahasannya.

Penentuan Case Types

Pertama-tama akan dibahas dulu pengertian dan perbedaan antara *case* dengan *case types*. *Case* merupakan sesuatu yang ada pada proses model, *case* dapat berupa aktivitas ataupun *task* pada proses model. Sedangkan *case types* merupakan

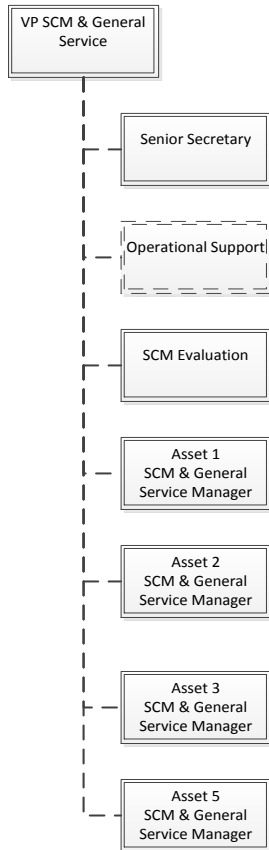
sesuatu yang mengalir pada proses model itu sendiri, dapat dikatakan juga bahwa *case types* merupakan *process instance* itu sendiri.

Untuk menentukan *case type* maka kita perlu mempertimbangkan apa yang menjadi *process instance* dari proses bisnis kasus ini. Sudah diketahui bahwa *process instance* dari kasus ini adalah ID karyawan. Melalui kesimpulan tersebut dapat diketahui juga bahwa jika ID karyawan berbeda maka jenis mobil yang diberikan juga akan berbeda, namun hal tersebut tidak akan mempengaruhi proses model yang akan berjalan, karena yang berbeda hanyalah jenis kendaraan yang diberikan saja, tidak terdapat perbedaan dalam pelayanan yang diberikan. Oleh karena itu dapat dilihat bahwa tidak terdapat perbedaan pelayanan pada proses model ini karena apapun jabatannya pelayanan yang diberikan pada proses model ini tetaplah sama yaitu menyediakan kendaraan bagi karyawan PT.PERTAMINA EP.

Dari analisis diatas dapat disimpulkan bahwa proses model ini hanya akan melihat *customer* sebagai satu jenis saja yaitu “pegawai” karena apapun jabatan yang dimiliki oleh karyawan yang melakukan *request*, pelayanan yang diberikan tetaplah sama yaitu menyediakan kendaraan bagi karyawan tersebut.

Penentuan *Business Function*

Untuk menentukan *business function* dapat dilihat struktur organisasi dari PT.PERTAMINA EP ini sendiri, karena proses bisnis reservasi kendaraan ini merupakan pekerjaan dari departemen SCM maka struktur organisasi yang dipakai adalah departemen SCM itu sendiri, berikut ini adalah struktur dari departemen SCM PT.PERTAMINA EP :



Gambar 2.2 Struktur organisasi Departemen SCM

Melalui struktur organisasi tersebut dapat dilihat bahwa divisi *Operational Support* merupakan divisi yang bertanggung-jawab terhadap kendaraan-kendaraan yang ada di PT.PERTAMINA EP. Dari divisi tersebut akan dibagi lagi menjadi beberapa ranah kerja, dan salah satu ranah kerjanya adalah transportasi PT.PERTAMINA EP, disini diketahui bahwa penganggung jawab atas pelayanan transportasi adalah seorang karyawan yang menjabat sebagai administrator supir. Selanjutnya, diketahui bahwa seorang karyawan PT.PERTAMINA EP jika ingin melakukan reservasi mereka

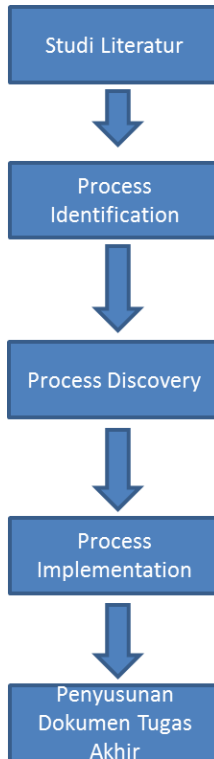
akan melakukan *request* terhadap admin supir, setelah itu supir akan memeriksa apakah terdapat mobil yang tersedia atau tidak, jika tidak maka akan disediakan *voucher* taksi bagi para karyawan yang melakukan *request* tersebut. Setelah itu setiap karyawan yang mendapatkan pelayanan transportasi kantor akan menggunakan kendaraan tersebut sesuai dengan kebutuhan mereka, setelah semua selesai maka setiap karyawan wajib untuk melakukan *reporting* bahwa mereka sudah selesai menggunakan kendaraan kantor tersebut, *reporting* dilaporkan kepada admin supir agar admin supir dapat melakukan pencatatan terhadap kendaraan yang sudah dipakai atau sedang dipakai.

Berdasarkan analisis diatas, dapat dilihat bahwa *business function* yang terbaik adalah beberapa bagian yang memang akan dilewati oleh karyawan (*customer*) untuk melakukan proses bisnis ini, bagian-bagian tersebut adalah admin supir, stok mobil, *voucher* taksi, dan pelaporan akhir, karena bagian-bagian tersebut-lah yang akan dilalui oleh karyawan untuk melakukan reservasi kendaraan kantor.

(halaman ini sengaja dikosongkan)

BAB III METODOLOGI PENELITIAN

Bab Metodologi ini akan menjelaskan mengenai tahapan pelaksanaan dari tugas akhir ini dan jadwal kegiatan dari tugas akhir. Metode yang akan digunakan untuk tugas akhir ini mengacu pada lifecycle dari business process management (BPM), berikut metodenya :



Gambar 3.1 Metodologi Tugas Akhir

3.1. Studi Literatur

Studi literature yang dipelajari adalah mengenai *business process management* (BPM) khususnya *software* yang akan digunakan pada tugas akhir kali ini yaitu BonitaSoft, selain itu

juga dilakukan *survey* lapangan mengenai kebutuhan PT.PERTAMINA EP terhadap *vehicle reservation systems* (VRS). Studi dilakukan dengan mempelajari pedoman buku *fundamental of business process management* dan juga beberapa *paper* yang membahas mengenai *business process management* dan berbagai macam penerapannya (implementasi). Penulis juga melakukan studi literature melalui video-video *tutorial* mengenai BonitaSoft.

3.2. *Process Identification*

Tahap ini adalah tahap identifikasi proses-proses yang ada pada aplikasi *vehicle reservation systems* (VRS). Hasil akhir dari tahap ini adalah sebuah *process landscape* yang didalamnya terdapat *case types* dan *business function*.

3.3. *Process Discovery*

Tahap ini akan dilakukan dengan cara melakukan *survey* pada PT.PERTAMINA EP, *survey* yang akan dilakukan adalah dengan *interview* dan *observation* terhadap karyawan-karyawan yang bersangkutan dengan proses bisnis aplikasi *vehicle reservation systems* dan terjun langsung ke lapangan untuk melihat praktek sebenarnya. Hasil akhir dari tahap ini adalah sebuah koreksi terhadap *process landscape* yang sudah dibuat pada tahap sebelumnya. Setelah itu dilanjutkan dengan pembuatan model dari *process landscape* tersebut. Hasil dari tahap ini adalah proses model dalam bentuk *as-is*.

3.4. *Process Implementation*

Tahap ini merupakan tahap terakhir yaitu pembuatan aplikasi menggunakan *bonitasoft*. Impelementasi ini akan menggunakan proses model *as-is* yang sudah dibuat pada *process discovery* sebelumnya. Hasil akhir dari *process implementation* ini adalah sebuah aplikasi reservasi kendaraan berbasis web yang dapat digunakan oleh karyawan PT.PERTAMINA EP.

3.5. Penyusunan Dokumen Tugas Akhir

Tahap ini adalah pengerjaan dokumen tugas akhir, dokumen ini akan diselesaikan dari bab 1 hingga bab 7, format pengerjaan dokumen tugas akhir ini akan disesuaikan dengan aturan-aturan yang sudah ditetapkan oleh Jurusan Sistem Informasi FTIF ITS Surabaya.

(halaman ini sengaja dikosongkan)

BAB IV PERANCANGAN

Pada bab empat ini akan membahas mengenai beberapa langkah-langkah awal dari BPM *lifecycle* yaitu *process discovery* dan perancangan implementasi pada BonitaSoft. Berikut ini akan dilakukan pembahasan dan penjelasan dari setiap tahapannya :

4.1. *Process Discovery*

Tahapan ini dilakukan dengan melakukan *survey* lapangan yaitu PT.PERTAMINA EP yang berlokasi di jl.dr.satrio jakarta selatan. *Survey* ini dilakukan selama 1 hari dengan metode *interview* dan *observation*.

Metode *interview* digunakan dengan melakukan wawancara berupa tanya-jawab kepada admin supir dari departemen SCM. Tanya-jawab ini dilakukan di ruangan admin supir tersebut ketika jam kerja. Hal-hal yang ditanyakan adalah mengenai *flow* dari alur reservasi kendaraan kantor bagi para karyawannya, mengenai dokumen-dokumen yang bersangkutan dan mengenai kebijakan-kebijakan yang berlaku pada perusahaan tersebut.



Gambar 4.1 Admin Supir Pertamina EP

Dari wawancara tersebut didapatkan hasil bahwa alur reservasi kendaraan kantor bagi para karyawannya memang diawali

dengan melakukan *request* kepada admin supir tersebut, setelah itu dilanjutkan dengan pengecekan oleh admin supir mengenai stok mobil yang tersedia berdasarkan jabatan karyawan yang bersangkutan, jika memang mobil tersedia maka admin supir akan memberikan *feedback* kepada karyawan yang melakukan *request* tersebut dan karyawan tersebut dapat menggunakan fasilitas kendaraan kantor, namun juga mobil yang bersangkutan tidak tersedia maka admin supir akan memberikan *voucher* taksi yang dapat digunakan oleh karyawan untuk menaiki taksi yang dianggap sebagai pengganti mobil kantor. Disini dapat dilihat bahwa tidak terdapat perubahan dari proses yang sudah dianalisis sebelumnya.

Selain itu wawancara ini juga menanyakan mengenai dokumen-dokumen yang digunakan pada proses reservasi kendaraan ini, hasilnya adalah tidak terdapat dokumen sama sekali yang digunakan pada proses reservasi ini karena memang proses reservasi ini hanya menggunakan email ataupun telepon untuk melakukan reservasinya, dokumen-dokumen pencatatan tidak dilakukan secara manual (hardcopy) namun dilakukan menggunakan IT, oleh karena itu tidak terdapat dokumen-dokumen cetak yang digunakan pada proses reservasi kendaraan kantor ini.



Gambar 4.2 Kondisi Basement Pertamina

Setelah itu wawancara juga menanyakan mengenai kebijakan yang berlaku dan mempengaruhi kebijakan proses reservasi

kendaraan ini. Terdapat beberapa kebijakan yang mempengaruhi proses ini, salah satunya adalah kebijakan mengenai jabatan dari karyawan yang melakukan bersangkutan tersebut. Jabatan dari karyawan berpengaruh terhadap tipe kendaraan yang dapat dipakai oleh karyawan yang melakukan *request* peminjaman kendaraan, tipe jabatan dikelompokkan menjadi 3 tipe yaitu staff, manager, dan VP / GM. Ketiga tipe karyawan tersebut mempunyai jatah tipe kendaraan yang berbeda-beda, staff mendapatkan jatah kendaraan bermerk toyota avanza, manager mendapatkan jatah kendaraan bermerk toyota innova dan VP / GM mendapat jatah kendaraan bermerk toyota fortuner, dan jika jatah mobil untuk jabatan tertentu sedang kosong dan terdapat *request* oleh karyawan tersebut maka admin supir akan memberikan *voucher* taksi yang dapat menggantikan fasilitas kendaraan kantor tersebut. Oleh karena itu kebijakan seperti ini akan mempengaruhi proses reservasi kendaraan.

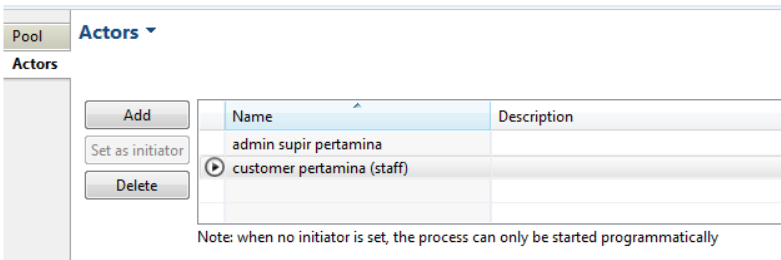
4.2. Business Process Model Notation

Pada sub-bab ini akan dibahas mengenai perancangan *tools* BonitaSoft yang akan digunakan untuk meng-implementasikan proses model *as-is* pada penelitian kali ini. Penggunaan *tools* BonitaSoft ini dapat dimulai dengan menetapkan proses model yang diterapkan pada studi kasus di tugas akhir ini, berikut ini akan ditampilkan pembuatan *business process model notation* (BPMN) yang akan digunakan.

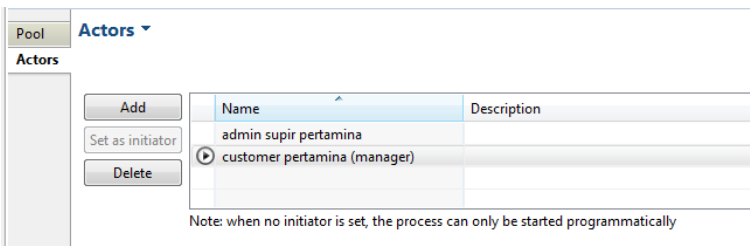
4.2.1 Penentuan Actor

Pertama-tama tentukan *actor* yang terlibat dalam studi kasus ini, terdapat 2 *actor* yang terlibat yaitu *customer* (*actor* yang berperan sebagai pemesan kendaraan, dalam studi kasus ini *customer* adalah karyawan PT.Pertamina EP yaitu staff, manager ataupun *Vice President*) dan admin supir (yaitu *actor* yang menjadi admin dari proses model ini, admin supir akan bertugas untuk melayani *request* dari *customer* dan bertanggung jawab atas semua kendaraan yang ada di

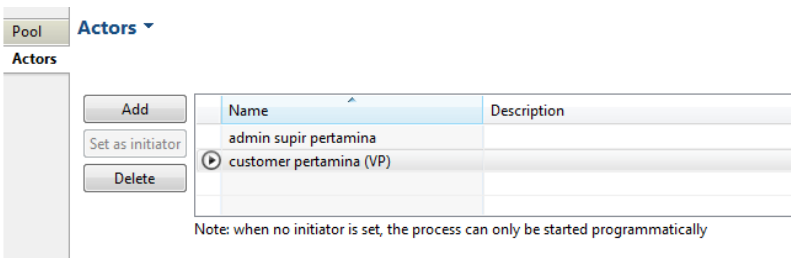
PT.Pertamina EP). Berikut ini adalah gambaran dari implementasi inisiasi *actor* pada aplikasi BONITA BPM :



Gambar 4.3 Actor untuk role staff



Gambar 4.4 Actor untuk role manager



Gambar 4.5 Actor untuk role VP

Pada gambar diatas dapat dilihat bahwa *actor* dibagi menjadi 3 bagian yaitu staff, manager dan VP, hal ini dilakukan karena terdapat perbedaan yang terhadap jabatan yang akan melakukan reservasi kendaraan kantor. Seperti yang sudah dibahas sebelumnya bahwa jabatan staff memiliki jatah mobil

avanza, jabatan manager memiliki jatah mobil innova sedangkan jabatan VP memiliki jatah mobil fortuner. Dengan adanya kebijakan tersebut maka *actor* pada BonitaSoft ini akan dibagi menjadi 3 (lebih tepatnya dibagi menjadi 3 *pool*, setiap *pool* akan mewakili satu jabatan).

Untuk membuat *actor* tersebut caranya cukup mudah, hanya perlu meng-klik tombol *add* dan memasukkan *name* dari *actor* yang diinginkan beserta *description* (tidak wajib diisi, hanya untuk keterangan tambahan saja). Setelah itu *actor customer* pertama (*customer* pada masing-masing *pool*) akan dipasang sebagai *initiator* yang artinya *actor* tersebut akan berperan untuk meng-inisiasi jalannya proses model ini, efeknya adalah ketika melakukan *run* aplikasi BONITA BPM ini maka yang dijalankan pertama kali adalah aplikasi yang diperankan oleh *initiator* yaitu *customer* pertama. Alasan kenapa *customer* pertama yang dijadikan *initiator* adalah karena *customer* pertama-lah yang akan memulai untuk membuat *request* pada proses model ini, sehingga proses model ini tidak akan berjalan jika tidak terdapat *request* dari *customer* pertama.

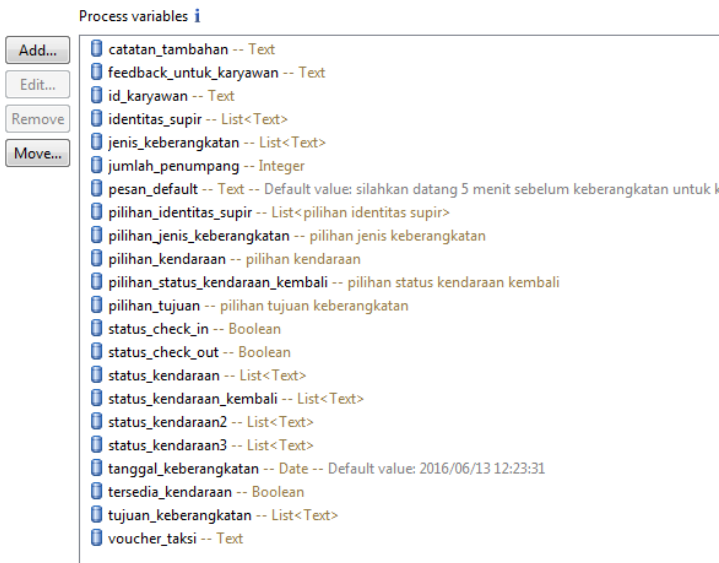
4.2.2 Pembuatan Proses Model

Setelah menetapkan *actor* yang terlibat, dapat ditentukan tugas-tugas atau *event* yang dilakukan oleh para *actor* tersebut. Namun sebelum membuat tugas atau *event* yang akan dilakukan oleh *actor*, akan lebih baik jika mendeklarasikan *variable* terlebih dahulu, karena *variable* akan menjadi elemen yang sangat penting pada peng-aplikasi-an tugas atau *event* dari BONITA BPM ini. Pembuatan *form* akan sangat bergantung pada *variable* yang telah disediakan sebelumnya. *Variable* akan dideklarasikan pada *pool* yang ada pada proses model ini, dapat dilihat pada gambar 4.6 adalah total dari *variable* yang telah dibuat pada proses model ini.

Penentuan *variable* bisa berdasarkan analisis dan perencanaan aplikasi yang akan dibuat menggunakan BONITA BPM ini. Setelah menentukan *variable* seperti gambar diatas maka dapat

dilanjutkan dengan membuat proses model yang nantinya akan dijadikan aplikasi sebagai bentuk dari *process implementation*.

Seperti yang sudah dibahas sebelumnya bahwa tugas dari *customer* dalam studi kasus ini adalah melakukan reservasi kendaraan kantor di PT.Pertamina EP, sedangkan tugas dari admin supir adalahanggapi atau melayani tanggapan yang sudah dibuat oleh *customer* tersebut.



Gambar 4.6 Variable yang digunakan

Berikut ini adalah urutan pekerjaan (*case*) yang akan dikerjakan oleh *customer* dan admin supir secara berbalasan :

1. *Customer* mengisi *form* (*form* yang dimaksud disini adalah *form request* untuk memakai atau meminjam kendaraan kantor untuk bepergian keluar untuk kepentingan PT.Pertamina EP).
2. Admin supir menanggapi *form* (yang dimaksud dengan “menanggapi” adalah memberikan *feedback* dengan menyatakan apakah terdapat kendaraan yang

tersedia untuk *request* yang sudah dimintai oleh *customer* tersebut ataupun memberikan catatan tambahan jika dirasa perlu).

3. Setelah itu akan terdapat percabangan (XOR) pada proses model ini, jadi terdapat 2 kemungkinan dan hal ini tergantung pada ketersediaan mobil yang di-*request* oleh *customer*. Jika terdapat mobil yang dapat memenuhi *request customer* maka proses model akan berjalan melalui cabang pertama, namun jika tidak terdapat mobil yang dapat memenuhi *request customer* maka proses model akan berjalan melalui cabang kedua (persediaan mobil akan bergantung pada jabatan yang dimiliki oleh *customer* yang melakukan *request* karena jatah mobil kantor yang dimiliki oleh setiap jabatan akan berbeda).
4. Pada cabang kedua/*decision* (tidak terdapat persediaan mobil untuk *customer*, percabangan kedua ini dapat juga disebut sebagai *negative outcome*) proses akan berlanjut untuk admin kembali, admin akan mendapat tugas untuk memasukkan *voucher* taksi yang akan digunakan bagi *customer* untuk melanjutkan pekerjaannya.
5. Setelah itu *customer* akan menerima *voucher* taksi yang telah dikirimkan oleh admin dan proses berakhir sampai disini (*negative outcome*).
6. Pada cabang pertama/*decision* (terdapat persediaan mobil untuk *customer*, percabangan pertama ini dapat juga disebut sebagai *positive outcome*) proses akan berlanjut untuk admin, admin akan mengisi data supir atau memploting supir yang akan bertugas untuk menanggapi *request customer* tersebut.
7. Setelah itu *customer* akan melakukan *pre check-in* yaitu membaca *feedback* yang berupa data *read-only* yang berisikan data-data *customer* (yang sebelumnya sudah diisi pada *form*) dan juga terdapat *feedback* dari admin untuk *customer* beserta identitas supir yang

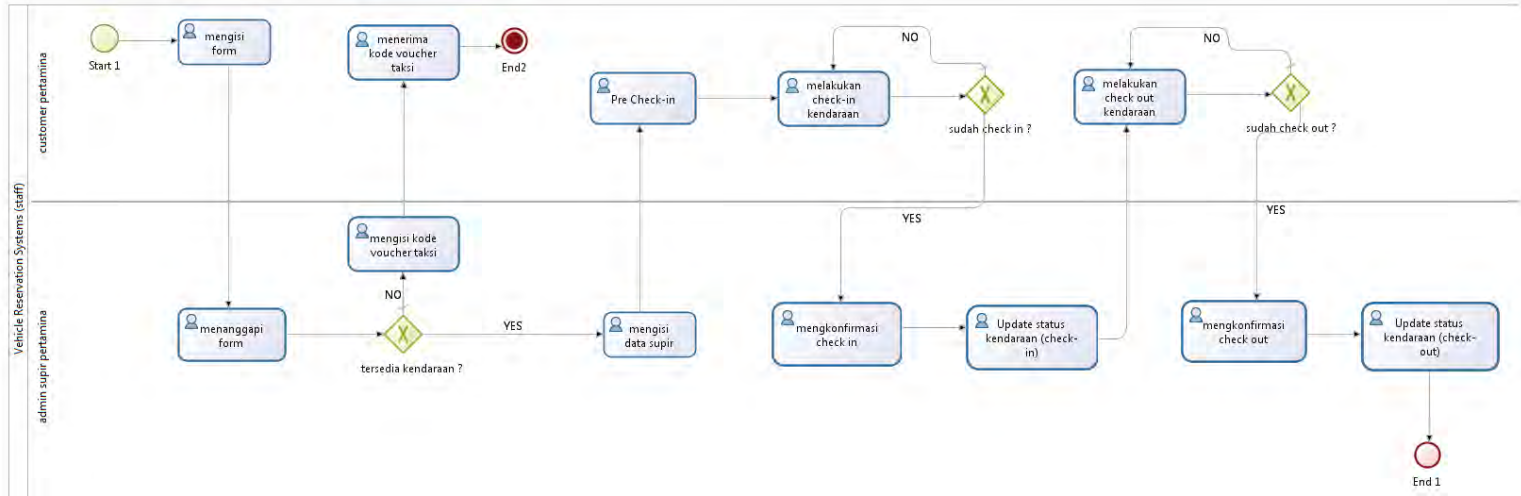
akan mengantarkan *customer* tersebut serta mobil yang akan digunakan oleh *customer*.

8. *Customer* yang sudah mendapatkan konfirmasi bahwa mereka sudah mendapatkan pinjaman kendaraan kantor akan melakukan *check-in* yang akan dilakukan sekitar 5-10 menit sebelum jam keberangkatan yang ditentukan, pada proses tersebut *customer* harus melakukan *check-in* yang berupa mencentang pernyataan pada aplikasi, pernyataan tersebut menyatakan bahwa *customer* telah mengambil kunci mobil dan sudah siap untuk berangkat bersama dengan supir yang bersangkutan.
9. Disini akan terdapat percabangan (decision) lagi, jika *customer* tidak mencentang pernyataan tersebut, maka sistem akan membawa *customer* kembali ke halaman *check-in* lagi, dan hal ini akan terus berulang (*looping*) sampai *customer* mencentang pernyataan tersebut.
10. Admin akan mengkonfirmasi *check-in* yang sudah dilakukan oleh *customer*, konfirmasi ini dilakukan hanya untuk melakukan pencatatan saja bagi admin supir.
11. Setelah itu admin supir akan mendapatkan tugas lagi yaitu melakukan *update* terhadap kondisi kendaraan yang *available* saat ini, akan terdapat *form* yang berisikan informasi mengenai kendaraan yang digunakan (ketika *customer* melakukan *check-in*) serta terdapat *check-box* yang berisikan *list* dari kendaraan keseluruhan, setelah itu admin supir akan mencentang kendaraan mana saja (pada *check-box*) yang masih *available* berdasarkan informasi yang didapat yaitu kendaraan yang sudah *check-in* (artinya kendaraan yang *check-in* akan berstatus *unavailable* dan tidak akan dicentang pada *check-box* oleh admin supir).
12. Setelah itu *customer* akan mengerjakan pekerjaannya menggunakan bantuan kendaraan kantor, dan kembali lagi ke kantor setelah pekerjaannya selesai. Ketika *customer* kembali ke kantor maka *customer* akan

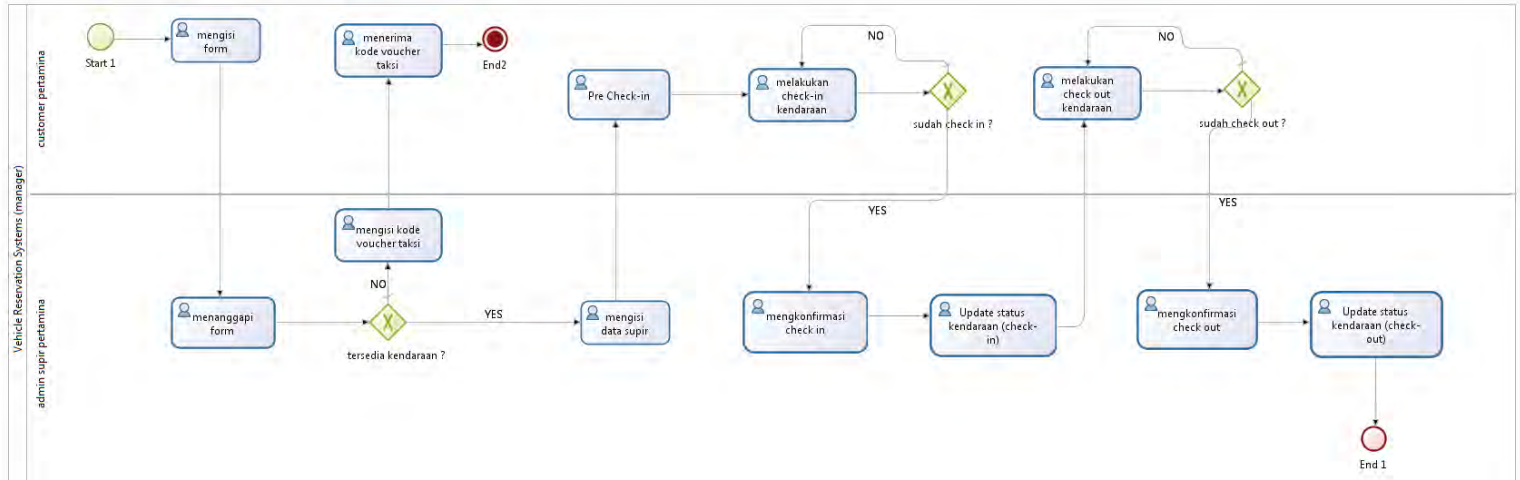
melakukan *check-out* yaitu proses pengembalian kendaraan kantor dan hal ini akan dilakukan dengan memberikan report kepada admin supir juga. *Report* tersebut berupa informasi mengenai kondisi kendaraan ketika sudah kembali ke kantor (disini terdapat beberapa pilihan kondisi yaitu “baik-baik saja”, “kecelakaan ringan” dan “kecelakaan berat”) serta mencentang pernyataan bahwa kendaraan sudah kembali ke kantor.

13. Disini akan terdapat percabangan (*decision*) lagi, jika *customer* tidak mencentang pernyataan tersebut, maka sistem akan membawa *customer* kembali ke halaman *check-in* lagi, dan hal ini akan terus berulang (*looping*) sampai *customer* mencentang pernyataan tersebut.
14. Setelah itu admin akan mengkonfirmasi laporan dari *custome*.
15. Dan terakhir, admin supir akan kembali melakukan *update* terhadap status kendaraan, karena terdapat kendaraan yang kembali (sudah melakukan *check-out*), akan terdapat *form* yang berisikan informasi serta *check-box* yang harus diisi oleh admin supir, informasi tersebut adalah kendaraan yang baru saja kembali (sudah melakukan *check-out*) dan *check-box* tersebut berisikan keseluruhan kendaraan yang ada dan harus dicentang oleh admin supir sehingga kendaraan yang *available* bisa ter-*update*.

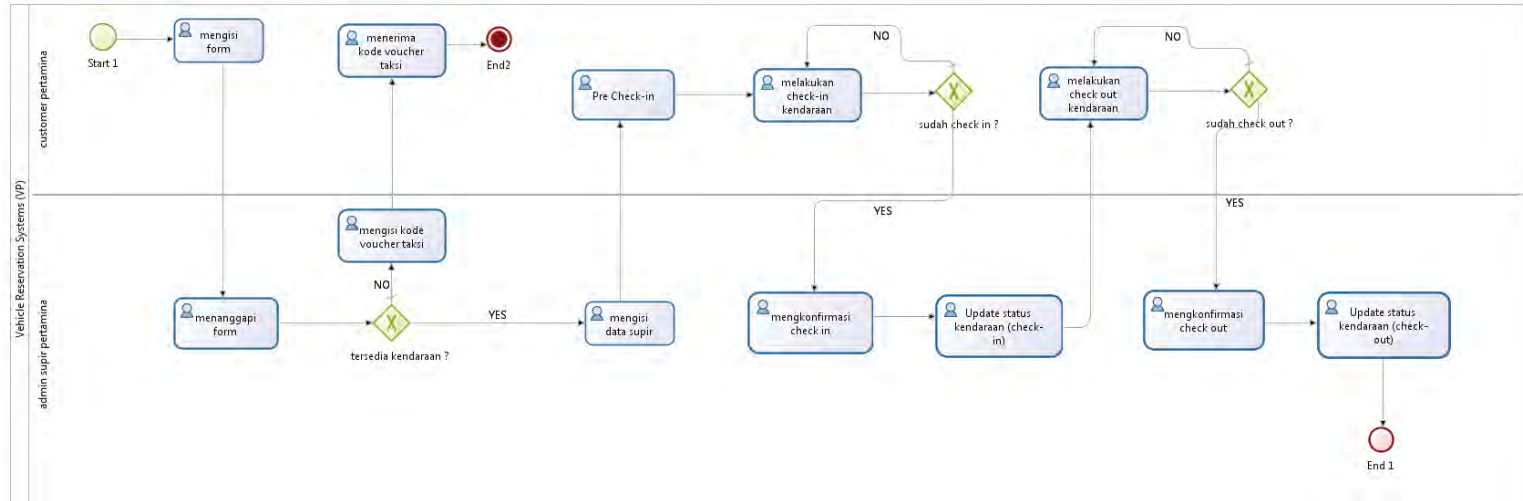
Setelah menetapkan proses yang berjalan seperti apa, maka tahap selanjutnya adalah membuat *business process model notation (BPMN)* yang akan digunakan sesuai dengan alur yang telah ditetapkan. Pada tahap inilah pembuat *BPMN* akan dilakukan menggunakan BONITA BPM, berikut ini adalah proses model yang dirancang :



Gambar 4.7 Proses Model as-is (role : staff)



Gambar 4.8 Proses Model as-is (role : manager)



Gambar 4.9 Proses Model as-is (role : VP)

Pada gambar 4.7, 4.8 dan 4.9 bisa dilihat bahwa terdapat 3 *pool* yang memiliki proses model yang sama namun memiliki *actor* yang berbeda (jabatannya). Sebenarnya *actor* pada proses model ini secara garis besar adalah sama yaitu *customer* PT.PERTAMINA EP, namun pada implementasi menggunakan BonitaSoft ini *actor* yang akan berperan menjadi *customer* terdapat 3 jenis yaitu staff, manager dan VP, yang nantinya akan mempengaruhi keputusan yang akan diambil pada proses model ini, sehingga harus dibuatkan *pool* yang berbeda untuk masing-masing *actor* yang akan berperan.

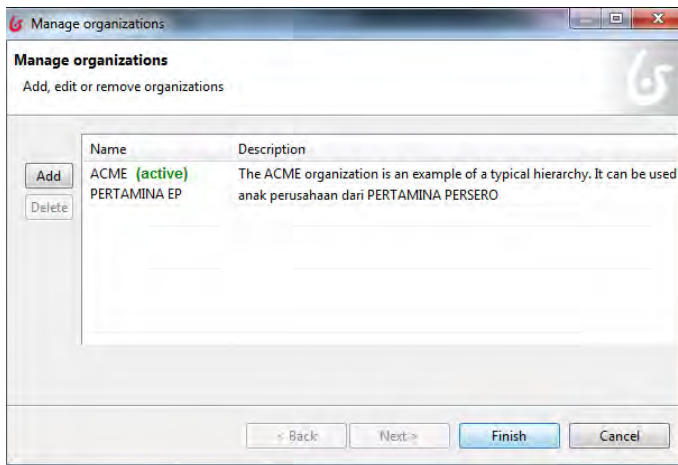
Dapat dilihat bahwa terdapat 2 *lane* yaitu *customer* pertamina dan admin supir pertamina, sehingga bisa disimpulkan bahwa setiap *lane* akan mewakili 1 *actor* yang terlibat dalam proses bisnis ini. Setelah itu terdapat beberapa *task* yang diwakilkan dengan gambar kotak, masing-masing *task* disini memiliki *form* yang akan menjadi tugas dari masing-masing *actor* pada *lane*-nya masing-masing. Sedangkan gambar belah-ketupat mewakili percabangan yang terdapat antara *task* satu dengan *task* lainnya, percabangan ini akan memiliki 2 jalan yang akan dipilih salah satu untuk meneruskan proses model ini (XOR), setiap percabangan akan bergantung pada *variable Boolean* yang dimiliki pada *task* sebelumnya yang ada pada proses model tersebut. Pada percabangan pertama yang bernama “tersedia kendaraan ?” nilai *variable Boolean* yang digunakan adalah “tersedia kendaraan” yang terdapat pada *task* “menanggapi *form*” yang merupakan *task* sebelum percabangan (XOR) tersebut. Setelah itu pada percabangan kedua yaitu “sudah *check-in*?” nilai *variable Boolean* yang digunakan adalah “status *check-in*” yang berada pada *task* sebelumnya yaitu “melakukan *check-in* kendaraan”. Percabangan terakhir yaitu “sudah *check-out*” memiliki nilai *variable Boolean* yaitu “status *check-out*” yang berada pada *task* sebelumnya yang bernama “melakukan *check out* kendaraan”. Jika diperhatikan *flow* utama dari proses model ini akan berjalan jika semua percabangan mempunyai nilai *true* pada percabangan (XOR) yang ada.

4.3. *Generate Bonita Software*

Setelah menentukan *actor* yang berperan serta membuat proses model yang akan dijalankan beserta *variable* dan *form* yang digunakan, maka tahap selanjutnya adalah *generate* atau *run* proses model yang sudah dibuat pada BONITA BPM tersebut. Namun sebelum melakukan hal tersebut ada beberapa hal yang perlu diperhatikan yaitu deklarasi organisasi dan *configure actor* yang sudah dibuat sebelumnya.

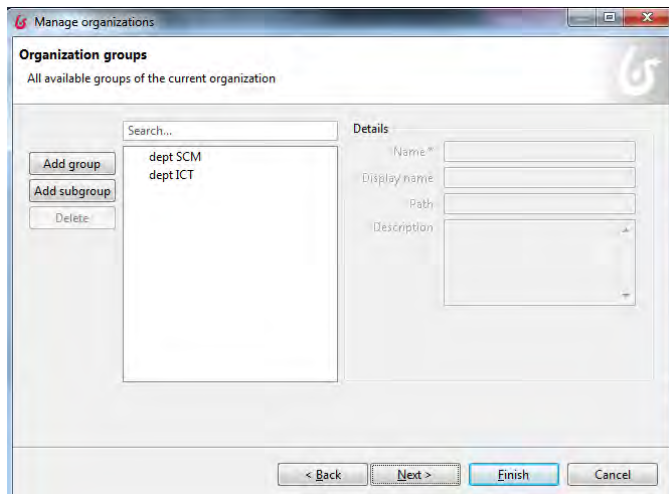
4.3.1. Deklarasi Organisasi

Pertama-tama akan dibahas dulu cara mendeklarasikan organisasi. Sebenarnya pada BONITA BPM terdapat organisasi *default* yang dapat digunakan untuk melakukan *run* proses model yang sudah dibuat, namun karena ini merupakan tugas akhir maka akan dibuat organisasi yang sesuai dengan studi kasus yang sedang dianalisis yaitu PT.PERTAMINA EP. Berikut ini adalah cara dan *screenshot* dari langkah-langkah deklarasi organisasi di BONITA BPM :



Gambar 4.10 Deklarasi Organisasi (langkah pertama)

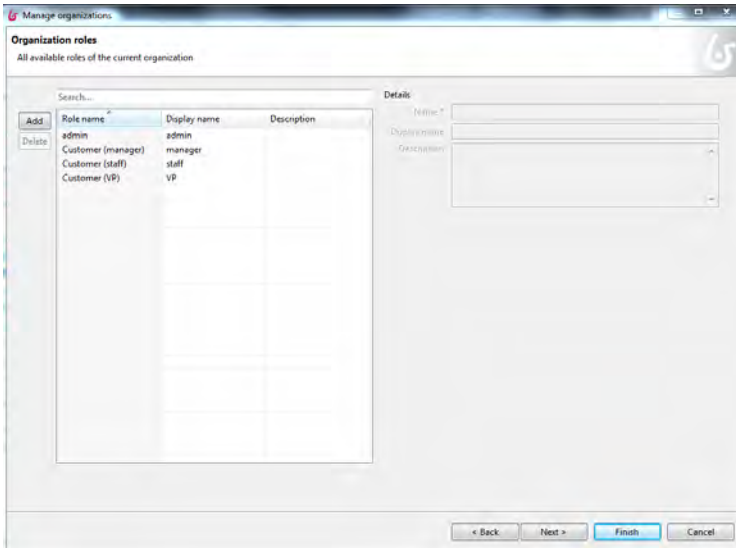
Gambar diatas merupakan tampilan dari *manage organization* yang ada pada BONITA BPM, pada gambar diatas sudah terdapat 2 organisasi yaitu ACME dan PERTAMINA EP (yang merupakan organisasi yang menjadi studi kasus pada tugas akhir ini), untuk membuat organisasi hanya perlu ditekan tombol “*add*” dan kemudian tekan tombol “*next*” (gambar diatas adalah kondisi dimana organisasi PERTAMINA EP sudah dibuat atau dideklarasikan pada BONITA BPM)



Gambar 4.11 Deklarasi Organisasi (langkah kedua)

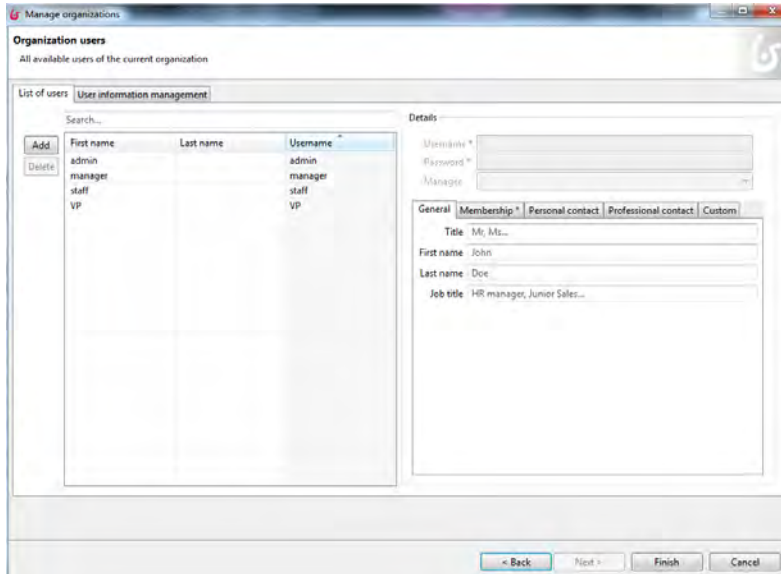
Selanjutnya akan muncul halaman *organization groups* yang merupakan halaman dimana kita bisa membuat struktur organisasi atau departemen-departemen apa sajakah yang terdapat pada organisasi tersebut, untuk menambahkan departemen klik “*add group*”, jika dirasa perlu ditambahkan sub-departemen klik “*add subgroup*”. Setelah itu jika dirasa sudah cukup maka klik *next*. Pada studi kasus ini terdapat 2 jenis departemen yang dimasukkan yaitu dept. SCM (*supply chain management*) dan dept ICT (*information and communication technology*), dept. SCM merupakan departemen yang bertanggung jawab atas kendaraan kantor

yang ada di PT.PERTAMINA EP, sedangkan dept. ICT merupakan departemen yang bertanggung jawab untuk melayani kebutuhan IT internal perusahaan (salah satunya adalah membuat aplikasi untuk keperluan internal perusahaan).



Gambar 4.12 Deklarasi Organisasi (langkah ketiga)

Setelah itu akan muncul halaman *organization roles* yang merupakan halaman dimana kita bisa membuat *roles* yang nantinya akan mendeskripsikan pekerjaan dari masing-masing akun (id) yang akan dibuat nantinya dan bisa *login* pada aplikasi yang dirancang pada BONITA BPM, *roles* disini juga bisa diartikan sebagai *actor* yang mempunyai peran dalam proses model yang sudah dirancang pada BONITA BPM. Untuk membuat *actor* tinggal klik *add* dan masukkan data yang diminta, setelah itu tekan tombol *next* untuk menuju ke halaman terakhir.

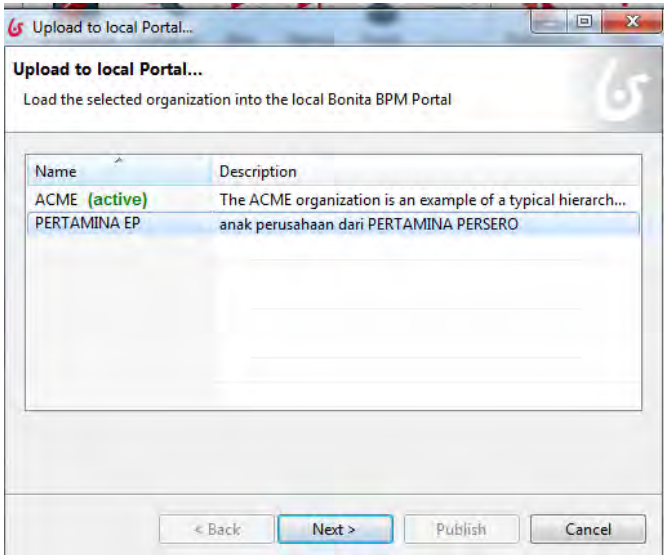


Gambar 4.13 Deklarasi Organisasi (langkah keempat).

Tahapan terakhir adalah pada halaman *organization users* yang merupakan halaman untuk membuat id untuk dapat login pada aplikasi BONITA BPM yang akan *run* nantinya. Pada saat ini sudah dibuatkan 6 id yaitu admin (yang akan digunakan untuk login sebagai admin supir), asst manager, manager, staff, VP (yang akan digunakan untuk login sebagai customer) dan id Joshua satriawan yang merupakan id dummy yang berperan sebagai ketua dari para *users* tersebut. Untuk membuat id tersebut hanya perlu klik *add* dan isi data yang diminta, jika sudah selesai klik tombol *finish* dan organisasi sudah selesai di deklarasikan.

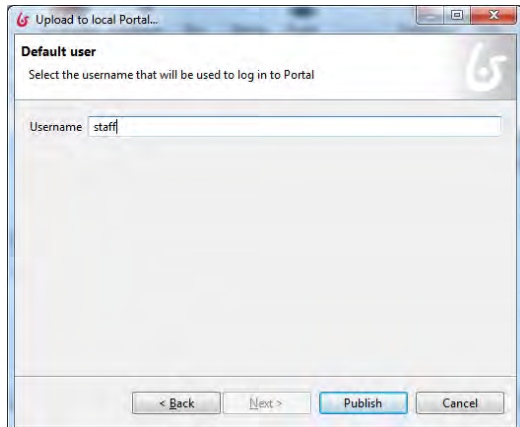
4.3.2. Publish Organization

Setelah mendeklarasikan organisasi PT.PERTAMINA EP maka langkah selanjutnya adalah mem-*publish* organisasi yang sudah dibuat tersebut pada BONITA BPM, berikut ini adalah langkah-langkah dan cara untuk *publish organization* :



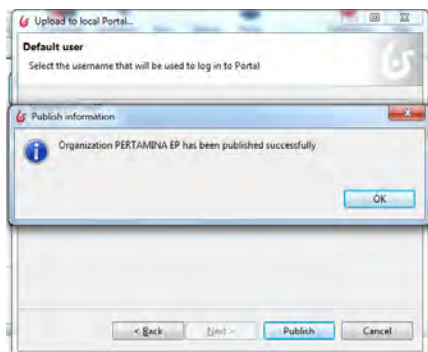
Gambar 4.14 Publish Organisasi (langkah pertama)

Pada menu pilih *organization > publish*, setelah itu akan tampil halaman “*upload to local portal*”, *local portal* adalah semacam *localhost* yang akan menampilkan aplikasi yang sudah dirancang pada BONITA BPM, disini hanya akan ada 1 organisasi yang dapat diaktifkan dalam 1 waktu. Pada gambar diatas terlihat bahwa organisasi yang aktif adalah ACME (terdapat tulisan *active* yang menandakan bahwa itu adalah organisasi yang sedang diaktifkan). Untuk mengaktifkan organisasi PERTAMINA EP, pilih organisasi tersebut dan klik tombol *next*.



Gambar 4.15 Publish Organisasi (langkah kedua)

Selanjutnya akan muncul halaman *default user* yang merupakan halaman yang akan mendeklarasikan user yang akan login ketika aplikasi pertama kali di *run* dari BONITA BPM, untuk kasus ini *user* pertama yang akan *login* adalah “staff”, cukup ketik staff pada kolom *username* dan klik *publish*.

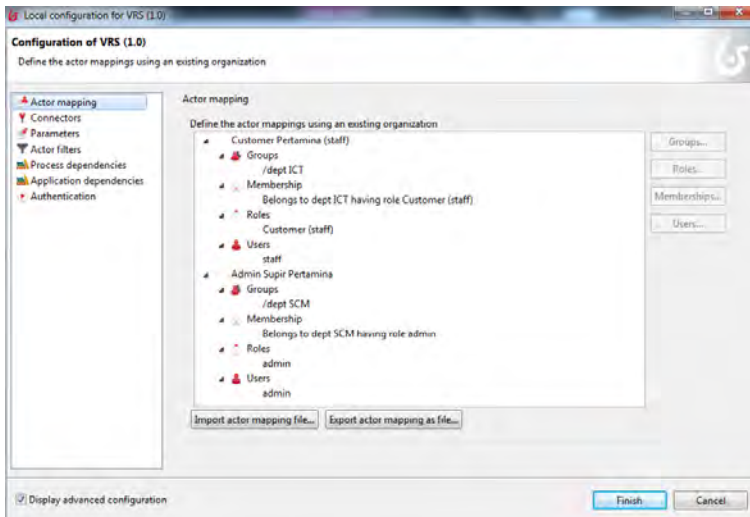


Gambar 4.16 Publish Organisasi (langkah ketiga)

Setelah itu akan muncul notifikasi seperti gambar diatas, hal ini menunjukkan bahwa organisasi PERTAMINA EP sudah di berhasil di-*publish*.

4.3.3. *Configure*

Setelah berhasil mendeklarasikan organisasi dan *publish* organisasi maka langkah selanjutnya adalah mengkonfigurasi *actor* yang akan berperan pada aplikasi ini berdasarkan *users* yang telah dideklarasikan pada saat mendeklarasikan organisasi PERTAMINA EP, berikut ini adalah caranya :



Gambar 4.17 *Configure* (actor mapping).

Pada menu bar pilih *configure*, setelah itu akan tampil halaman *configuration* dari proses model yang sedang dirancang pada BONITA BPM. Pada gambar diatas dapat dilihat terdapat 2 *actor* yang digunakan pada proses model ini yaitu *customer* pertamina dan *admin supir pertamina*, namun kedua *actor* tersebut belum dimasukkan ke dalam organisasi PERTAMINA EP, oleh karena itu kedua *actor* tersebut harus memilih 4 komponen yang terdapat dalam organisasi pada aplikasi BONITA BPM ini yaitu *groups*, *memberships*, *roles*, dan *users*. *Groups* merupakan departemen dimana para *actor* tersebut akan berperan, *roles* merupakan peran yang akan dimainkan oleh para *actor* tersebut (dalam studi kasus ini

terdapat 2 *roles* yaitu admin dan customer), *membership* merupakan gabungan dari *groups* dan *roles* yaitu deklarasi departemen tempat *actor* tersebut akan bekerja dan *roles* atau peran yang akan dimainkan oleh *actor* tersebut, sedangkan yang terakhir adalah *users* yaitu akun yang akan digunakan oleh *actor* tersebut untuk *login* pada aplikasi yang sudah dirancang pada BONITA BPM. Setelah selesai melakukan semua itu klik *finish* untuk keluar dari halaman *configuration*.

(halaman ini sengaja dikosongkan)

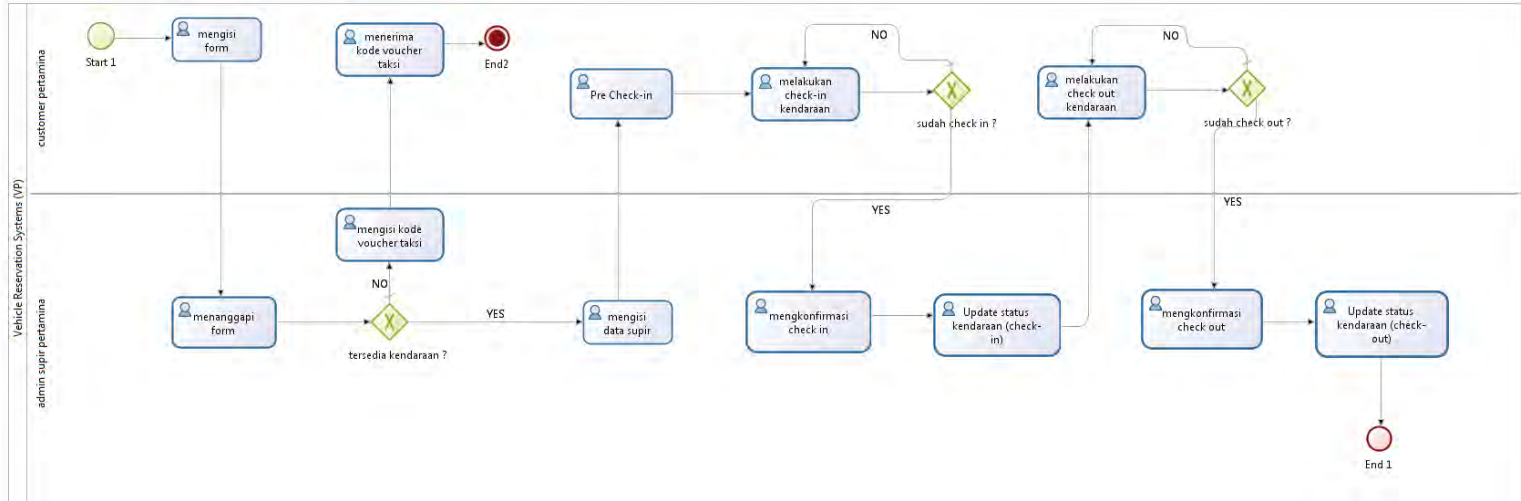
BAB V IMPLEMENTASI

Pada bab lima ini akan membahas *process implementation* menggunakan BonitaSoft, *process implementation* yang akan dibahas adalah mengenai langkah-langkah implementasi untuk membuat aplikasi berbasis web sesuai dengan studi kasus menggunakan BonitaSoft. Berikut ini akan dilakukan pembahasan dan penjelasan dari setiap tahapannya :

5.1 Membuat *Form*

Pada sub-bab ini akan dibahas langkah-langkah untuk membuat *form* pada BPMN yang sudah tersedia. Ditunjukkan pada gambar 5.1 proses model *as-is* yang sudah dibuat sebelumnya.

Pada gambar 5.1 dapat dilihat bahwa terdapat terdapat 12 *case* pada proses model *as-is* yang sudah dibuat berdasarkan studi kasus penelitian kali ini. Masing-masing *case* tersebut memiliki *form* yang berbeda-beda berdasarkan aktifitasnya masing-masing, sebagai contoh pada *case* pertama yaitu “mengisi *form*”, pada *case* tersebut memakai beberapa *variable* yang sudah di-deklarasikan sebelumnya, Ditunjukkan pada gambar 5.2 adalah *variable* yang dipakai pada *case* pertama yaitu “mengisi *form*”.



Gambar 5.1. Proses Model *as-is*

Business variables		Process variables	
<input type="checkbox"/> Select all			
Name	Widget	Mandatory	Read only
<input checked="" type="checkbox"/> id_karyawan	Text field	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> tujuan_keberangkatan	List	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> jumlah_penumpang	Text field	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> jenis_keberangkatan	List	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> tanggal_keberangkatan	Date	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> catatan_tambahan	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_tujuan	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_jenis_keberangkatan	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> tersedia_kendaraan	Checkbox	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> feedback_untuk_karyawan	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> voucher_taksi	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pesan_default	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_identitas_supir	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> identitas_supir	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_check_in	Checkbox	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan_kembali	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_status_kendaraan_kembali	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_check_out	Checkbox	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_kendaraan	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan2	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan3	List	<input type="checkbox"/>	<input type="checkbox"/>

Gambar 5.2. Add form (ceklist variable) pada case "mengisi form"

Pada gambar 5.2 terdapat 6 *variable* yang dicentang yang artinya 6 *variable* tersebutlah yang digunakan pada *case* “mengisi form”. Dapat dilihat juga bahwa terdapat 5 *variable* yang dicentang kolom *mandatory* yang artinya *variable* tersebut harus diisi oleh *user* (dalam kasus ini *user* adalah *customer* PT.PERTAMINA EP).

Setelah memilih *variable* tersebut maka akan muncul halaman untuk *edit form* , dapat dilihat pada gambar 5.3 adalah tampilannya :

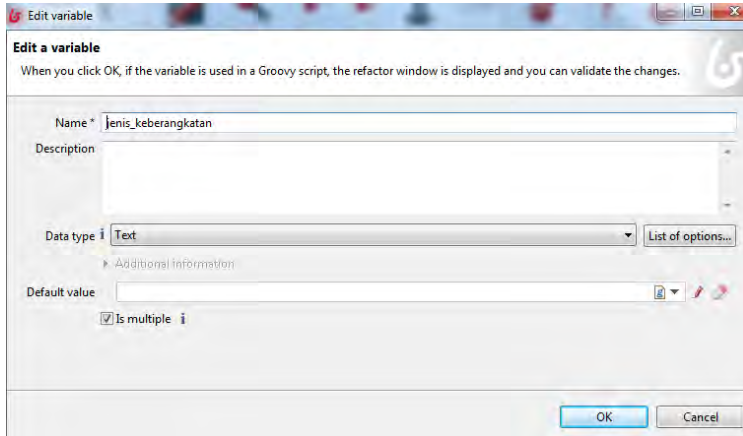
The image shows a web form titled "Edit Form" for a travel application. The form is divided into several sections by dashed lines. The first section contains two input fields: "Id karyawan" and "Tujuan keberang...". The second section contains two input fields: "Jumlah penumpang..." and "Jenis keberangkat...". The third section contains a date input field labeled "Tanggal keberan..." with the value "01 / 01 / 2010". The fourth section contains a text input field labeled "Catatan tambahan". At the bottom of the form is a button labeled "SEND TO ADMIN".

Gambar 5.3. Edit Form pada case "mengisi form"

Gambar 5.3 merupakan halaman *edit form* pada case “mengisi form”, *variable-variable* yang muncul pada halaman ini adalah *variable* yang sudah dipilih sebelumnya, dan jenis tampilan masing-masing *variable* juga berdasarkan *type variable* yang sudah dipilih ketika deklarasi *variable* sebelumnya. Dapat dilihat pada gambar diatas terdapat 6 *variable* yang ada pada halaman ini, *variable* tersebut merupakan *variable* yang sudah dicentang pada langkah sebelumnya.

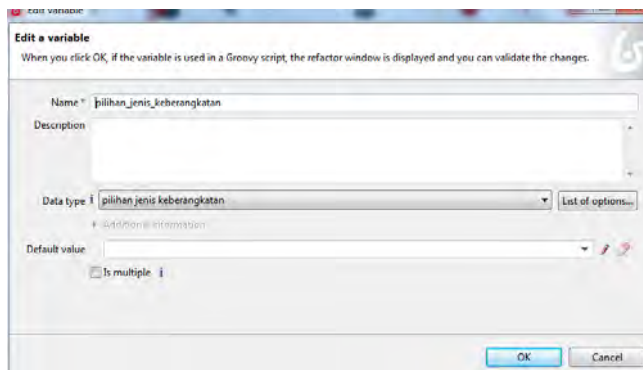
5.1.1. Memasukkan *available values*

Pada beberapa *variable* terdapat *variable* yang memiliki nilai *multiple* yang artinya *variable* tersebut mempunyai nilai yang beragam. *Variable* ini di-deklarasikan memiliki nilai *multiple* agar dapat dimasukkan beberapa nilai (lebih dari satu) dan menjadi pilihan (*options*) bagi *user* (dalam kasus ini adalah *customer* PT.PERTAMINA EP) yang menggunakan aplikasi ini. Berikut ini contohnya :



Gambar 5.4. Properti dari Variable "jenis_keberangkatan"

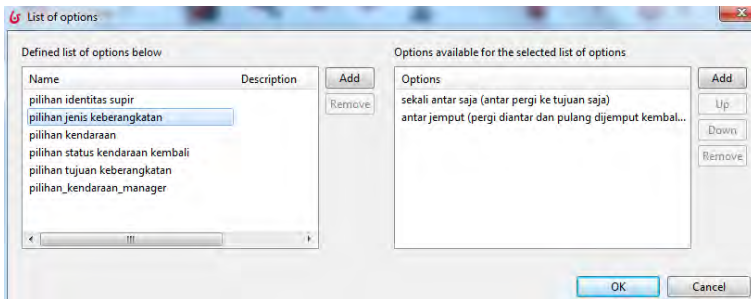
Gambar 5.4 adalah info dari *variable* "jenis_keberangkatan" yang memiliki *type text* dan mempunyai nilai *multiple*. Ketika *variable* memiliki nilai *multiple* maka harus terdapat pula *variable* yang memiliki bertujuan untuk mengisi nilai *multiple* tersebut, berikut ini adalah contohnya :



Gambar 5.5 Property dari Variable "pilihan_jenis_keberangkatan"

Gambar 5.5 adalah gambar dari *variable* "pilihan_jenis_keberangkatan", *variable* ini di-deklarasikan agar dapat mengisi nilai *multiple* dari *variable*

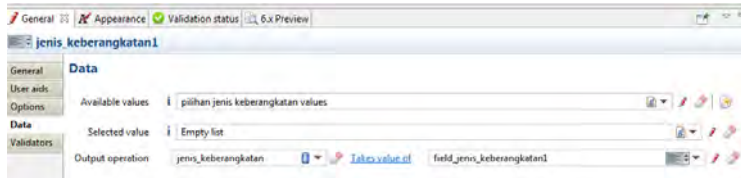
“jenis_keberangkatan”, dapat dilihat bahwa *data type* dari *variable* ini adalah “pilihan jenis keberangkatan” yang artinya *data type* tersebut dibuat sendiri dan memiliki nilai yang ditujukan untuk mengisi nilai *multiple* pada *variable* “jenis_keberangkatan”, berikut ini adalah isi dari nilai “pilihan jenis keberangkatan” (nilai ini akan mengisi nilai *multiple* dari *variable* “jenis_keberangkatan”) :



Gambar 5.6. Pilihan dari *variable* "pilihan_jenis_kendaraan"

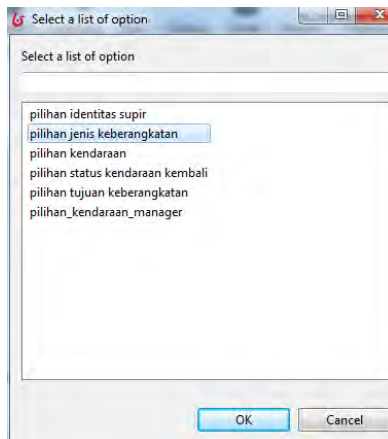
Pada gambar 5.6 dapat dilihat bahwa terdapat 2 nilai yaitu “sekali antar saja” dan “antar jemput”, kedua nilai inilah yang nantinya akan mengisi nilai *multiple* pada *variable* “jenis_keberangkatan”.

Untuk mengisi nilai *multiple* maka pada halaman *edit form*, pilihlah *variable* “jenis_keberangkatan”, setelah itu pada tab *general* pilih kolom *data*



Gambar 5.7. Property dari Variable "jenis_keberangkatan" pada halaman edit form

Pada kolom *data* tersebut lihatlah pada kolom *available values* dan klik tombol yang paling kanan yaitu *generate expression from a list of option*, setelah itu akan muncul tampilan seperti ini :

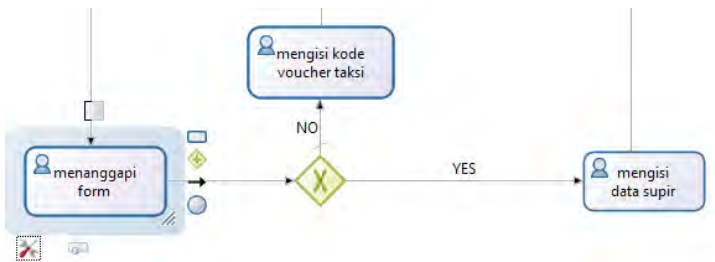


Gambar 5.8. Pilihan untuk mengisi nilai *multiple* pada variable "jenis_keberangkatan"

Pada gambar 5.8 ini pilihlah *variable* “pilihan jenis keberangkatan”. Setelah itu klik tombol OK dan akhirnya nilai *multiple* pada *variable* “jenis_keberangkatan” sudah terisi dengan 2 *option* yaitu “antar jemput” dan “sekali antar saja”.

5.1.2. Menggunakan *decision XOR*

Pada beberapa *variable* terdapat *variable* yang mempunyai nilai *Boolean*, *variable* ini bertujuan untuk membuat percabangan (*decision*) pada proses model yang akan dibuat. Sebagai contoh pada *case* “menanggapi form” akan digunakan salah satu *variable* yang memiliki nilai *Boolean* yaitu *variable* “tersedia_kendaraan”. *Variable* ini dibuat untuk menyatakan apakah tersedia kendaraan atau tidak untuk *user* (*customer* PT.PERTAMINA EP) yang sudah melakukan *request*.



Gambar 5.9. Decision XOR

Gambar diatas merupakan sepotong gambar dari proses model yang sudah dibuat, gambar diatas menunjukkan bahwa setelah *case* “menanggapi form” akan terdapat *decision XOR* yang artinya proses akan berjalan ke salah satu arah dari percabangan tersebut (dapat ke arah “YES” sehingga berjalan menuju *case* “mengisi data supir” dan dapat ke arah “NO” sehingga berjalan menuju *case* “mengisi kode voucher taksi”).

Berikut ini akan dibahas cara menggunakan *variable Boolean* dan menghubungkannya dengan *decision XOR* :

Name	Widget	Mandatory	Read only
<input checked="" type="checkbox"/> id_karyawan	Text field	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> tujuan_keberangkatan	List	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> jumlah_penumpang	Text field	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> jenis_keberangkatan	List	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> tanggal_keberangkatan	Date	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> catatan_tambahan	Text field	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pilihan_tujuan	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_jenis_keberangkatan	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> tersedia_kendaraan	Checkbox	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> feedback_untuk_karyawan	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> voucher_taksi	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pesan_default	Text field	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_identitas_supir	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> identitas_supir	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_check_in	Checkbox	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan_kembali	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_status_kendaraan_kembali	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_check_out	Checkbox	<input type="checkbox"/>	<input type="checkbox"/>
<input checked="" type="checkbox"/> status_kendaraan	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> pilihan_kendaraan	Radio buttons	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan2	List	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> status_kendaraan3	List	<input type="checkbox"/>	<input type="checkbox"/>

Gambar 5.10. Add form (ceklist variable) pada case "menanggapi form"

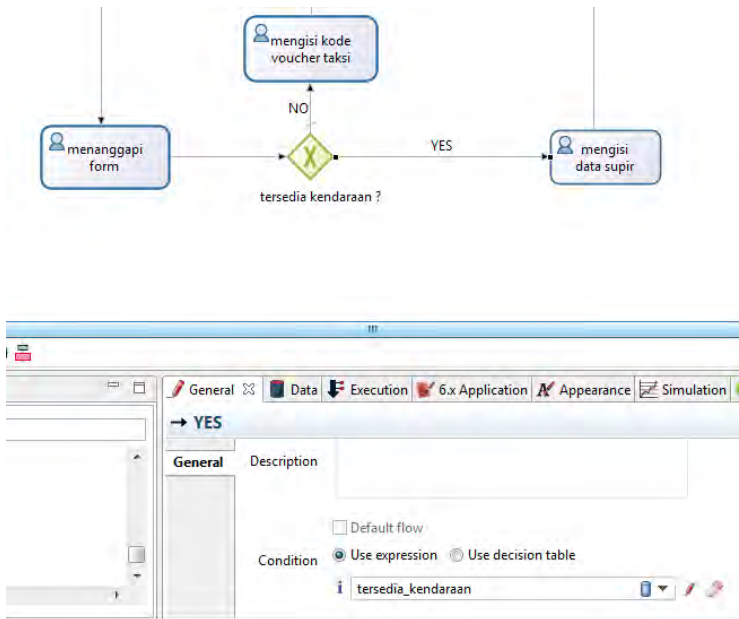
Gambar 5.10 merupakan gambar dari *variable* yang dipilih untuk mengisi *form* dari *case* “menanggapi *form*”. Dapat dilihat bahwa *variable* “tersedia_kendaraan” (merupakan *variable* dengan nilai *Boolean*) dicentang sehingga akan masuk ke dalam halaman *edit form*, berikut tampilannya :

Gambar 5.11. Halaman *edit form* pada case "menanggapi form"

Gambar 5.11 merupakan tampilan dari halaman *edit form* pada *case* “menanggapi *form*”, dapat dilihat bahwa terdapat

variable “tersedia_kendaraan” pada halaman ini. *Variable* tersebut ditampilkan dalam bentuk *checkbox* yang artinya dapat dicentang ataupun tidak dicentang. Hal inilah yang akan membuat *variable* tersebut memiliki nilai *true or false*, ketika *variable* tersebut dicentang maka nilai yang akan keluar adalah *true* sedangkan ketika *variable* tersebut tidak dicentang maka nilai yang akan keluar adalah *false*.

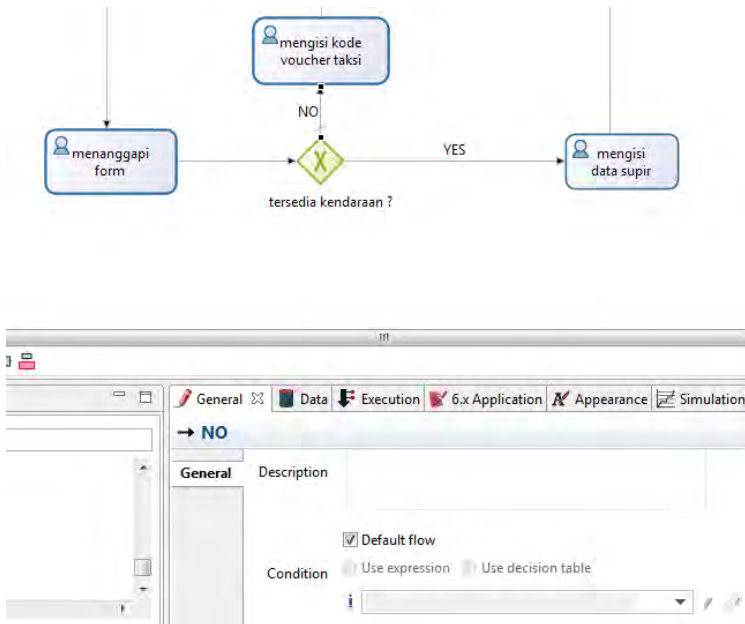
Setelah itu cara mendeklarasikan bahwa nilai *true* akan menuju *case* “mengisi supir” sedangkan nilai *false* akan menuju *case* “mengisi voucher taksi” adalah seperti berikut :



Gambar 5.12. Property pada garis "YES"

Gambar diatas adalah tampilan *property* dari garis “YES”, dapat dilihat bahwa pada *property* > tab *general* dan pada kolom *condition* terdapat pilihan *use expression*, jika kolom tersebut dipilih maka artinya *decision XOR* jika bernilai *true*

akan berjalan menuju garis tersebut (dalam kasus ini adalah garis “YES”), dan pada kolom dibawahnya terdapat kolom untuk mengisi *variable* yang gunanya adalah untuk menentukan nilai dari *variable* manakah yang akan diambil nilai *true or false*-nya (dalam kasus ini yang akan diambil adalah nilai dari *variable* “tersedia_kendaraan”).



Gambar 5.13. Property pada garis "NO"

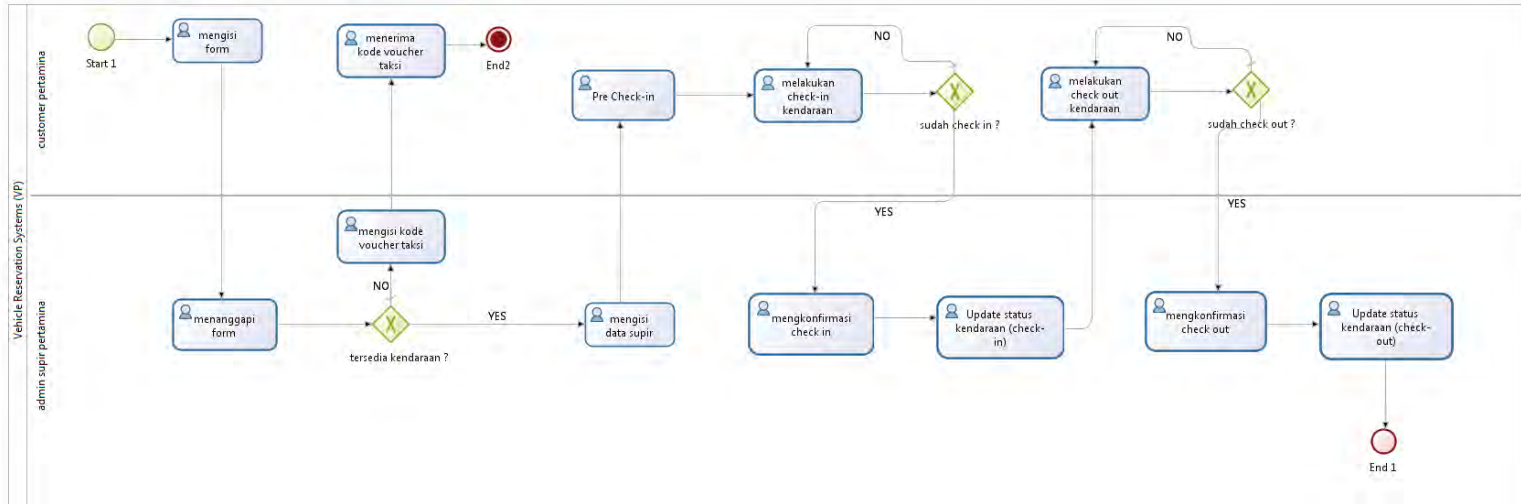
Sedangkan pada gambar diatas ini menampilkan *property* dari garis “NO” , pada gambar diatas cukup mencentang kolom *default flow* yang artinya proses akan berjalan ke arah garis tersebut jika nilai *Boolean* memiliki nilai *false*.

5.1.3. Membuat *Terminate and End* dan *End*

Pada BPMN hanya akan terdapat satu *positive outcome* dan boleh terdapat lebih dari 1 *negative outcome*. Pada studi kasus

kali ini akan terdapat 1 *positive outcome* dan 1 *negative outcome*. Ditunjukkan pada gambar 5.14 yaitu proses model *as-is* secara keseluruhan.

Pada gambar 5.14 dapat dilihat bahwa terdapat 1 lambang *end* dan 1 lambang *terminate and end*. Lambang *end* merupakan *positive outcome* sedangkan lambang *terminate and end* merupakan lambang *negative outcome*. *Positive outcome* akan mengakhiri proses dengan hasil yang terbaik (merupakan hasil yang ditujukan untuk proses ini pada awalnya) sedangkan *negative outcome* merupakan hasil yang tidak diinginkan / alternatif. Dalam studi kasus ini *positive outcome* akan mengakhiri proses dimana terdapat kendaraan untuk *user* (*customer* PT.PERTAMINA EP) yang melakukan *request*. Sedangkan *negative outcome* akan mengakhiri proses ketika tidak terdapat kendaraan yang dapat memenuhi *request* dari *user* yang pada akhirnya akan diberikan *voucher* taksi sebagai gantinya.



Gambar 5.14. Proses Model *as-is*

(halaman ini sengaja dikosongkan)

BAB VI HASIL DAN PEMBAHASAN

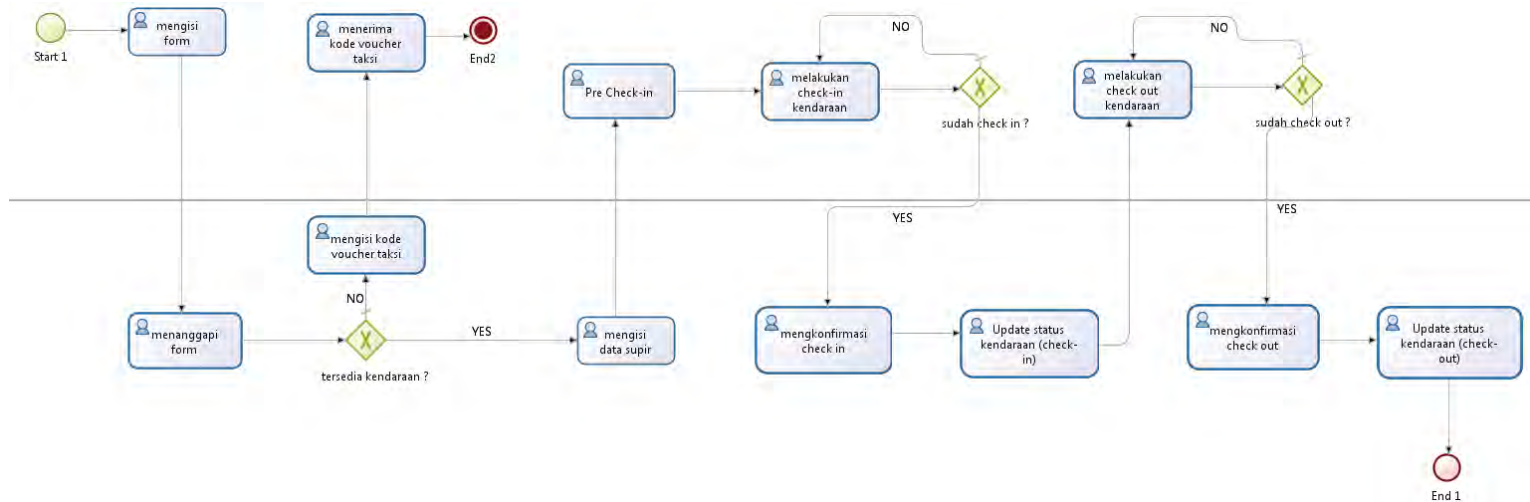
6.1 Permodelan *as-is* Sistem Reservasi Kendaraan PT.PERTAMINA EP

Dalam tugas akhir kali ini, permodelan *as-is* untuk sistem reservasi kendaraan di PT.PERTAMINA EP dibuat menggunakan pendekatan proses bisnis manajemen (*business process management*, atau disebut BPM). Dapat dilihat pada gambar 6.1 hasil dari permodelan *as-is* dari sistem reservasi kendaraan di PT.PERTAMINA EP :

6.2 Automasi Sistem Reservasi (*as-is*) PT.PERTAMINA EP

Dalam tugas akhir kali ini, automasi sistem reservasi menggunakan model *as-is* di PT.PERTAMINA EP adalah dalam tahapan *process implementation*, lebih tepatnya lagi adalah menggunakan *tools* yaitu BonitaSoft. Dibawah ini akan dibahas dan ditampilkan hasil dari *process implementation* (automasi) tersebut.

Setelah membuat proses model seperti diatas pada BonitaSoft, dan melakukan pengaturan (*publish organization, configure, declare variable, dan run*) maka akan muncul tampilan BONITA PORTAL, berikut ini adalah pembahasannya ;



Gambar 6.1. Hasil Permodelan *as-is* Sistem Reservasi Kendaraan PT.PERTAMINA EP

- Gambar 6.2 adalah tampilan dari *case* “mengisi *form*”, *form* ini akan diisi oleh *customer* (staff, Manager atau VP, namun dalam contoh di gambar yang mengisi *form* adalah staff), terdapat satu *text-box* yang bersifat *read-only* yaitu “id karyawan” karena *text-box* ini memiliki nilai yang sudah pasti (yaitu id dari karyawan yang sedang login), setelah itu sisanya adalah *text-box* dan *list-box* yang bersifat *mandatory* (wajib diisi) kecuali *text-box* yang terakhir (paling bawah) yaitu “catatan tambahan”, kolom tersebut tidak wajib untuk diisi.

Gambar 6.2. Tampilan dari *case* "mengisi *form*"

- Gambar 6.3 adalah tampilan dari *case* “menanggapi *form*” yang akan diisi oleh admin supir setelah terdapat *customer* yang melakukan *request* untuk kendaraan kantor. Terdapat beberapa kolom yang bersifat *read-only* yaitu kolom-kolom yang sudah diisi oleh *customer* sebelumnya, setelah itu terdapat juga 3 kolom yang akan dilihat dan diisi ataupun dibiarkan oleh admin supir yaitu “tersedia kendaraan” (*Boolean*, bernilai *true or false*), “kendaraan *available* dan akan digunakan” (*check-box*, tidak bersifat *mandatory*), serta “*feedback* untuk karyawan” (bersifat *mandatory*).

Bonitasoft Welcome: admin

Tasks Cases Processes

Menanggapi form

Id karyawan
staff123

Jumlah penumpang
1

Tanggal keberangkatan
2016 July 24

Catatan tambahan
test

Tersedia kendaraan
☒

Feedback untuk karyawan
test

Tujuan keberangkatan
PERTAMINA PERSERO

Jenis keberangkatan
sekalu-antar

Kendaraan available yang dapat digunakan
avanza 1
avanza 2
avanza 3
avanza 4

CONTINUE

Gambar 6.3. Tampilan dari case "menanggapi form"

- Gambar 6.4 ditampilkan *form* yang akan muncul ketika tidak tersedia kendaraan untuk memenuhi *request* yang dilakukan oleh *customer*. *Form* yang muncul adalah *form* untuk mengisi *voucher* taksi yang akan dilakukan oleh admin supir. *Form* berisikan dua kolom yaitu kolom "id karyawan" yang bersifat *read only* berisikan id yang sudah diisi oleh *customer* ketika melakukan *request* dan kolom kedua adalah *voucher* taksi yang diisi oleh admin supir untuk menggantikan kendaraan yang tidak tersedia.

Bonitasoft Welcome: admin User Settings

Tasks Cases Processes

Mengisi Voucher Taksi

Id karyawan
pertamindo123

Voucher taksi

SEND TO CUSTOMER

Gambar 6.4. Tampilan dari case "mengisi voucher taksi"

- Setelah itu proses akan ditujukan untuk *customer* yang melakukan *request* sebelumnya, *form* yang ditampilkan berisikan dua kolom yang bersifat *read-*

only yaitu “id karyawan” dan “voucher taksi” yang sudah diisi oleh admin supir sebelumnya.

The screenshot shows the Bonitasoft web application interface. At the top, there's a navigation bar with 'Tasks', 'Cases', and 'Processes' tabs. The main title is 'Menerima Kode Voucher Taksi'. Below the title, there are two input fields. The first field is labeled 'id karyawan' and contains the text 'partemasa123'. The second field is labeled 'Voucher taksi' and contains the text 'voucher123'. At the bottom right of the form, there is a button labeled 'SEND'.

Gambar 6.5. Tampilan dari case "menerima kode voucher taksi"

- Gambar 6.6 akan ditampilkan *form* yang akan muncul ketika tersedia kendaraan pada saat admin supir menanggapi *request* yang dilakukan oleh *customer*. Selanjutnya admin supir akan mengisi atau memploting supir yang bertugas untuk menanggapi *request* tersebut. Terdapat 3 kolom yang bersifat *read-only* dan terdapat 1 kolom yang berupa *list-box* yang bersifat *mandatory*. Terdapat “id karyawan”, “kendaraan yang digunakan” serta “tersedia kendaraan” (*Boolean*) yang bersifat *read-only*, sedangkan terdapat 1 *list-box* yaitu “identitas supir” yang akan dipilih oleh admin supir untuk menanggapi *request* dari *customer*.

The screenshot shows the Bonitasoft web application interface. At the top, there's a navigation bar with 'Tasks', 'Cases', and 'Processes' tabs. The main title is 'Mengisi Data Supir'. Below the title, there are four input fields. The first field is labeled 'id karyawan' and contains the text 'partemasa123'. The second field is labeled 'Tersedia kendaraan' and contains a checkbox. The third field is labeled 'Kendaraan yang digunakan' and contains a dropdown menu with the value 'bus1234'. The fourth field is labeled 'Identitas supir' and contains a list box with the following items: 'supir 1 - andung akbar', 'supir 2 - sulherman', 'supir 3 - rahadiyan dandar', and 'supir 4 - aditya mayapada'. At the bottom right of the form, there is a button labeled 'SEND TO CUSTOMER'.

Gambar 6.6. Tampilan dari case "mengisi data supir"

- Setelah itu *form* selanjutnya akan ditujukan kepada *customer*, *form* ini akan bersifat *read-only* karena pada *form* ini *customer* hanya akan membaca (*review*) dari *request* hingga *feedback* yang dilakukan oleh admin supir.

Bonitasoft Welcome: staff User Settings

Tasks Cases Processes

Membaca Feedback

Id karyawan
Pertamina123

Jumlah penumpang
1

Tanggal keberangkatan
2016 June 13

Tersedia kendaraan
☐

Feedback untuk karyawan
Test tersedia kendaraan

Identitas supir
supir 8 - aktifnya masyarakat

Kendaraan yang digunakan
kendaraa 1

Pesan default
silahkan datang 5 menit sebelum keberangkatan untuk konfirmasi check-in

Done

Gambar 6.7. Tampilan dari case "membaca feedback"

- Setelah itu *customer* akan melakukan *check-in* (keberangkatan), *form* yang ditujukan kepada *customer* adalah *form* yang berisikan kolom-kolom *read-only* (*customer* dapat membaca info-info yang sudah diisi sebelumnya) dan juga terdapat satu kolom yang berisikan pernyataan (bernilai *Boolean*) yaitu “saya sudah mengambil kunci mobil dan siap berangkat bersama supir”, kolom ini akan diisi jika *customer* sudah melakukan *check-in* pada *pool* yang tersedia di PT.PERTAMINA EP, jika *customer* tidak mencentang pernyataan tersebut maka *form* akan *looping* dan kembali lagi pada *form* ini.

Bonitasoft Welcome: staff User Settings

Tasks Cases Processes

Melakukan Check In kendaraan

Id karyawan
Pertamina123

Jumlah penumpang
1

Tanggal keberangkatan
2016 June 13

Identitas supir
supir 8 - aktifnya masyarakat

Kendaraan yang digunakan
kendaraa 1

saya sudah mengambil kunci mobil dan siap berangkat bersama supir
☐

CHECK IN

Gambar 6.8. Tampilan dari case "melakukan check-in kendaraan"

- Setelah itu admin supir akan melakukan konfirmasi *check-in*, *form* yang ada pada aktivitas ini bersifat *read only* karena admin supir cukup membaca (*review*) terhadap *check-in* yang dilakukan oleh *customer*.

Gambar 6.9. Tampilan dari case "mengkonfirmasi *check-in*"

- Setelah itu admin supir akan mendapatkan *task* lagi yaitu melakukan *update* kendaraan karena baru saja terdapat kendaraan yang melakukan *check-in*. Pada *form* ini terdapat dua kolom yaitu kolom *list box* yaitu "kendaraan yang digunakan" yang bersifat *read only*, selain itu terdapat juga kolom *check-box* yang akan dicentang oleh admin supir dengan kendaraan-kendaraan yang masih *available* pada saat tersebut.

Gambar 6.10. Tampilan dari case "update status kendaraan *check-in*"

- Setelah itu *customer* akan melakukan pekerjaannya menggunakan kendaraan dan akan kembali lagi ke PT.PERTAMINA EP, berikutnya *customer* akan melakukan *check-out*, disini akan terdapat 5 kolom,

dimana 3 kolom diantaranya akan bersifat *read-only* sedangkan 1 kolom lainnya bersifat *mandatory* dan 1 kolom lainnya akan diisi oleh *customer*.

Bonitasoft Welcome: staff User Settings

Tasks Cases Processes

Melakukan Check Out Kendaraan

Id karyawan
Pertamina123

Identitas supir
siyah k - satriya masyuda

Kendaraan yang digunakan
benz c

Status kendaraan kembali
baik-baik saja kendaraan ringan kendaraan parah

kendaraan sudah kembali ke pool PEPertamina EP

RECORD

Gambar 6.11. Tampilan dari case "melakukan *check-out* kendaraan"

- Selanjutnya admin supir akan mengkonfirmasi *check-out* yang sudah dilakukan oleh *customer*. Konfirmasi *check-out* ini akan berbentuk *form* yang bersifat *read-only* seluruhnya, disitu akan tercantum info mengenai id karyawan, identitas supir, kendaraan yang digunakan, status kendaraan dan pernyataan dari *customer* ketika melakukan *check-out* sebelumnya.

Bonitasoft Welcome: admin User Settings

Tasks Cases Processes

Mengkonfirmasi Check Out

Id karyawan
Pertamina123

Identitas supir
siyah k - satriya masyuda

Kendaraan yang digunakan
benz c

Status kendaraan kembali
baik-baik saja

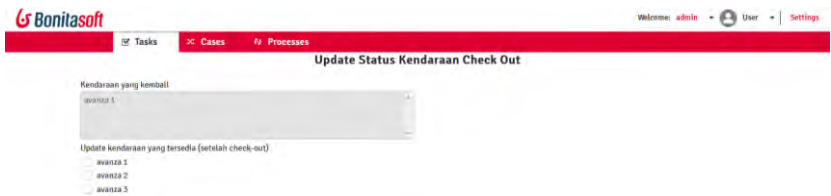
Status check-out

START

Gambar 6.12. Tampilan dari case "Mengkonfirmasi *check-out*"

- Dan terakhir, admin supir akan melakukan *update* status kendaraan setelah *customer* melakukan *check-out*, disini akan terdapat 2 kolom yaitu “kendaraan yang kembali” (bersifat *read only*, berisikan informasi mengenai kendaraan yang baru saja kembali ke kantor), dan kolom “update kendaraan yang tersedia (setelah *check-out*)” yang akan diisi oleh admin supir

berdasarkan kendaraan-kendaraan yang *available* pada saat itu.



Gambar 6.13. Tampilan dari case "update status kendaraan check-out"

(halaman ini sengaja dikosongkan)

LAMPIRAN A

Lampiran A berisikan Proses model *To-Be* yang disertai dengan *process analysis*, dan *process redesign*

Pada lampiran ini akan dibahas mengenai analisis untuk membuat sebuah proses model *to-be*. Proses model *to-be* ini merupakan hasil analisis dari *process analysis* dan *process redesign*. Berikut ini pembahasannya :

1.1 Process Analysis

Pada *process analysis* ini metode yang akan digunakan adalah *qualitative analysis* tepatnya menggunakan *value added analysis table*, berikut pembahasannya :

1.1.1 Qualitative Analysis

Sebelum memasuki penilaian pada *qualitative analysis*, akan dilakukan pembahasan mengenai penentuan *customer* pada studi kasus ini. Studi kasus ini membahas mengenai reservasi kendaraan kantor pada PT.PERTAMINA EP, dapat diketahui bahwa yang melakukan reservasi dan menikmati fasilitas kendaraan kantor adalah karyawan dari PT.PERTAMINA EP itu sendiri, dengan catatan karyawan-karyawan tersebut tidak melakukan pembayaran atau tidak mengeluarkan biaya sedikitpun, dan dapat ditarik kesimpulan yang mengeluarkan biaya adalah kantor (perusahaan) PT.PERTAMINA EP itu sendiri. Selanjutnya dapat diketahui pula bahwa tujuan dari peminjaman kendaraan kantor ini adalah untuk keperluan kantor saja. Oleh karena itu *customer* dari studi kasus ini adalah PT.PERTAMINA EP itu sendiri karena yang mengeluarkan biaya untuk melakukan proses ini adalah PT.PERTAMINA EP, dan proses ini juga dilakukan hanya untuk keperluan perusahaan.

1.1.1.1 Value Added Analysis

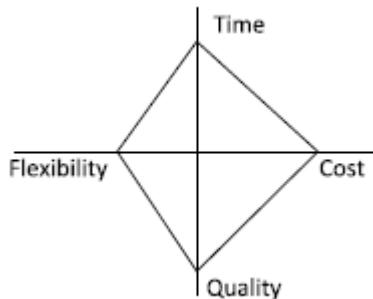
Pada *value added analysis* akan dilihat dan dianalisis melalui *case* dari proses model *as-is* yang sudah terbentuk, setiap *case* akan diberikan nilai berdasarkan analisis penulis. Jenis nilai akan dibedakan menjadi 3 jenis yaitu *value-adding* (VA), *business value-adding* (BVA) dan *non-value adding* (NVA). Berikut ini pembahasannya : .

Table A.9.1 Value-added table

Step	Performer	Classification
Mengisi Form	Karyawan	VA
Menanggapi Form	Admin Supir	BVA
Mengisi Kode Voucher Taksi	Admin Supir	NVA
Menerima Kode Voucher Taksi	Karyawan	VA
Mengisi Data Supir	Admin Supir	NVA
Pre Check-in	Karyawan	NVA
Melakukan Check-in Kendaraan	Karyawan	NVA
Mengkonfirmasi Check-in	Admin Supir	BVA
Update Status Kendaraan (Check-in)	Admin Supir	BVA
Melakukan Check-out Kendaraan	Karyawan	NVA
Mengkonfirmasi Check-out	Admin Supir	BVA
Update Status Kendaraan (Check-out)	Admin Supir	BVA

1.2 Process Re-Design

Proses *re-design* ini membahas suatu konsep yang dinamakan *devil's quadrangle* yang memiliki seperti :



Gambar 4. 1 Devil Quadrangle

Pada studi kasus ini, penulis akan memberikan suatu proses model yang dapat disarankan bagi perusahaan PT.Pertamina EP sebagai proses model yang lebih efektif dan efisien, dengan catatan penulis akan memilih salah satu dari kriteria *devil's quadrangle* tersebut untuk bisa dipilihkan dan diterapkan sebagai salah satu kategori yang akan diperbaiki / diutamakan untuk proses model yang baru.

Proses model yang dibahas pada tugas akhir kali ini adalah mengenai reservasi kendaraan kantor pada PT.Pertamina EP, permasalahan yang dialami oleh perusahaan saat ini terletak pada sumber daya manusianya (SDM) karena SDM tidak menunjukkan partisipasinya dalam mendukung proses bisnis yang sedang berjalan ini, contoh nyata permasalahannya adalah, pada saat karyawan kembali ke kantor bersama dengan supir dan kendaraan kantornya seharusnya karyawan tersebut memberikan laporan kepada admin supir bahwa kendaraan kantor sudah kembali sehingga supir dapat membuat record bahwa kendaraan kantor yang bersangkutan sudah kembali dan *available* kembali. Namun yang terjadi adalah ketika para karyawan tersebut kembali tidak semua karyawan melaporkan

bahwa mereka sudah kembali dan kendaraan juga sudah kembali, hanya supir saja yang melaporkan bahwa kendaraan sudah kembali, padahal admin supir memerlukan *feedback* tidak hanya dari supir namun dari karyawan juga.

1.2.1 Initiate

Pada tahap ini yang akan dilakukan adalah menentukan akan codong kearah manakah proses model *to-be* yang akan dirancang pada tugas akhir kali ini. Terdapat 4 perspektif yaitu *time*, *cost*, *quality* dan *flexibility*. Permasalahan yang dihadapi terletak terhadap SDM, seperti yang sudah dibahas sebelumnya bahwa SDM tidak terlalu dapat diajak bekerja sama untuk menjalankan proses yang berjalan sehingga terjadi penundaan proses yang cukup lama. Oleh karena itulah dibutuhkan proses model yang baru dengan memperbaiki dari sisi waktu (*time*). Selain itu diperlukan juga perbaikan dari sisi biaya (*cost*) karena memang dengan adanya masalah pada SDM ini membuat proses ini menjadi panjang dengan tingkat kepentingan yang tidak *urgent* namun harus dilakukan sedangkan terdapat *cost* yang dikeluarkan walaupun *cost* tersebut tidak besar. Oleh karena itu pada penelitian kali ini akan difokuskan untuk memperbaiki *time* dan *cost* pada proses model ini.

Setelah itu, dilakukan juga analisis terhadap *heuristics process redesign* yang mempunyai 7 *heuristics* dimana masing-masing *heuristics* juga mempunyai kriterianya masing-masing, dengan total 26 kriteria pada 7 *heuristics* tersebut.

1.2.2 Design

Tahap selanjutnya adalah mencari nilai sub-kategori mana sajakah yang bisa diterapkan untuk memperbaiki nilai *cost* dan *time* pada proses model ini, terdapat 7 sub-kategori yang dapat memberikan peningkatan pada nilai *cost* dan *time* yaitu :

- *Integration*
- *Case types*

- *Activity elimination*
- *Activity composition*
- *Resequencing*
- *Activity automation*
- *Interfacing*

Berikut ini merupakan tabel penilaian positif (+) dan negatif (-) serta netral (.) pada sub-kategori tersebut :

Table A9.2. 7 heuristics table

	<i>Time</i>	<i>Cost</i>	<i>Quality</i>	<i>Flexibility</i>
<i>integration</i>	+	+	.	-
<i>Case Types</i>	+	+	-	-
<i>Activity Elimination</i>	+	+	-	.
<i>Activity Composition</i>	+	+	.	-
<i>Resequencing</i>	+	+	.	.
<i>Activity Automation</i>	+	+	-	-
<i>Interfacing</i>	+	+	.	-

Berikut ini adalah penjelasan teknis mengenai apa yang harus dilakukan dengan 7 *heuristics* yang akan diterapkan untuk membuat proses model *to-be* pada *process redesign* ini :

- *Integration*
Menganalisis proses yang dapat diintegrasikan antar *actor* yang terlibat dalam proses model, jika terdapat proses tersebut maka akan terdapat proses yang digabungkan atau mungkin ditambahkan tergantung dengan studi kasus.
- *Casetypes*
Mendefinisikan dan menganalisis apakah aktivitas-aktivitas yang ada pada proses model sudah cocok dengan *casetypes* yang ada pada proses model tersebut. Jika tidak cocok maka aktivitas yang bersangkutan akan membuat proses bisnis yang baru.
- *Activity elimination*

Menghapus aktivitas yang tidak dibutuhkan pada proses model, cara yang paling mudah adalah dengan mengeliminasi aktivitas dengan nilai NVA (*non value adding*)

- *Activity composition*
Menggabungkan aktivitas “kecil” dengan aktivitas “kecil” lainnya sehingga menjadi sebuah aktivitas yang lebih besar sehingga dapat mengurangi waktu pengerjaan proses.
- *Resequencing*
Memindahkan aktivitas ke tempat yang lebih baik jika aktivitas tersebut tidak dibutuhkan untuk *follow-up* aktivitas selanjutnya atau menunda aktivitas yang memang dirasa bisa ditumpuk dan dimaksimalkan untuk dilaksanakan nantinya. Hal ini akan menghemat pengeluaran dan mempercepat waktu proses.
- *Activity automation*
Mengotomatisasi proses dengan menggunakan sistem atau teknologi sehingga mengurangi pekerjaan yang dilakukan oleh manusia, hal ini akan berefek untuk mempercepat waktu.
- *Interfacing*
Heuristics ini menyarankan agar *interface* atau tampilan dari proses model ataupun aplikasi yang digunakan dapat dipertimbangkan sesuai dengan kebutuhan *user* ataupun para pekerjanya, tujuannya agar dapat mempercepat pelaksanaan proses pelaksanaan.

1.2.3 Evaluate

Pada tahapan ini akan dibuatkan proses model yang baru berdasarkan analisis terhadap *heuristics* yang sudah dibahas sebelumnya. Kriteria pada *devil's quadrangle* yang akan ditingkatkan adalah *cost* dan *time*, terdapat 7 *heuristics* yang mendukung kriteria *cost* dan *time*. Berdasarkan 7 *heuristics* tersebut maka hal-hal yang perlu dilakukan adalah :

- Desain *interface* sesuai dengan kebutuhan dan keinginan *actor* dari proses model tersebut.
- Menganalisis apakah terdapat proses yang dapat diotomatisasi menggunakan sistem atau teknologi sehingga mengurangi campur tangan manusia.
- Menganalisis proses model apakah terdapat urutan aktivitas yang bisa dihapus atau ditunda tanpa mengganggu jalannya proses bisnis.
- Menganalisis apakah terdapat beberapa aktivitas yang dapat digabungkan menjadi satu aktivitas.
- Menghapus aktivitas yang mempunyai nilai NVA.
- Menganalisis apakah aktivitas yang bersangkutan dapat membuat proses bisnis yang baru dengan *casetypes* yang berbeda.
- Menganalisis apakah proses model dapat menerapkan integrasi sehingga antar *actor* terdapat komunikasi untuk mendukung hasil yang optimal dari proses model.

1.3 Pembahasan Proses Model *To-Be*

Pada bab sebelumnya sudah dibahas bahwa terdapat 7 *heuristics* yang akan meningkatkan nilai *cost* dan *time* pada *devil's quadrangle*, namun tidak seluruh 7 *heuristics* tersebut dapat diterapkan pada proses model *as-is* yang ada, akan ada pertimbangan dan analisis terkait dengan penerapan proses model *to-be* dalam realitinya pada studi kasus yang digunakan. Berikut ini adalah pembahasan mengenai *heuristics* yang digunakan penerapannya oleh penulis dalam membuat proses model *to-be*:

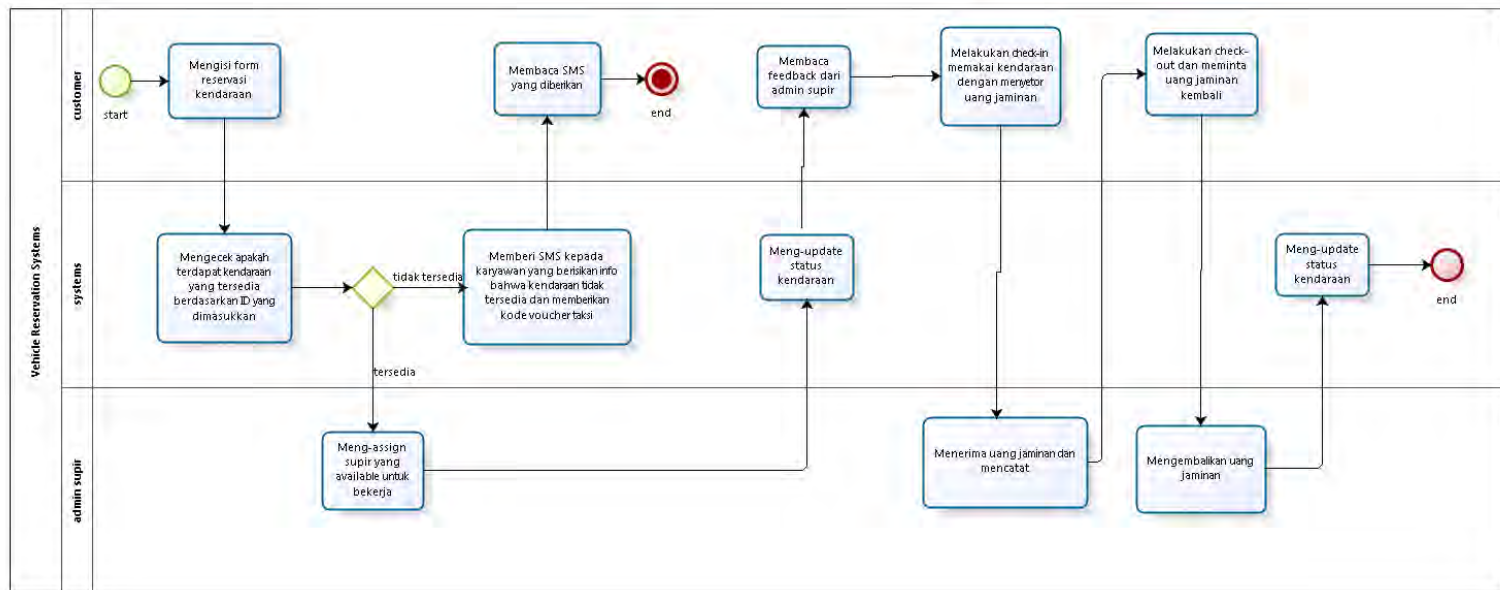
Table A9.3. penerapan *heuristics* pada proses model *to-be*

<i>Heuristics</i>	Penerapan
<i>Interfacing</i>	<ul style="list-style-type: none"> • Membuat desain <i>interface</i> yang sama antar <i>customer</i> dengan admin supir sehingga mencegah kesalah pahaman

	<p>dalam komunikasi.</p> <ul style="list-style-type: none"> • Melakukan <i>testing</i> aplikasi kepada <i>user</i> (<i>customer</i> dan admin supir)
<i>Activity Elimination</i>	<ul style="list-style-type: none"> • Mengganti aktivitas <i>NVA</i> dengan aktivitas yang dapat lebih otomatis, <i>NVA</i> tersebut adalah : <ul style="list-style-type: none"> • Mengirimkan data tersebut melalui email kepada admin supir (<i>NVA</i> ini diganti dengan aktivitas mengisi <i>form</i> dan mengklik tombol <i>submit</i> untuk mengirim <i>form</i> tersebut) • Karyawan membuka email dan membaca <i>voucher</i> taksi yang diberikan (aktivitas ini diganti dengan menerima SMS yang berisikan <i>voucher</i> taksi).
<i>Activity Composition</i>	<ul style="list-style-type: none"> • Tidak dapat diterapkan, karena aktivitas yang ada pada proses model ini memiliki tujuan yang berbeda-beda sehingga tidak dapat digabungkan, jika dipaksa untuk digabungkan maka proses model tidak akan berjalan.
<i>Case Types</i>	<ul style="list-style-type: none"> • Tidak dapat diterapkan, karena tidak ditemukan <i>special case</i> pada proses model ini sehingga tidak perlu ada proses model lain yang dibuat.
<i>Resequencing</i>	<ul style="list-style-type: none"> • Tidak dapat diterapkan karena proses model tergolong pendek. • Tidak dapat diterapkan karena urutan

	<p>aktivitas yang ada sudah efektif (sehingga tidak dapat diubah lagi urutannya atau dipindahkan, jika diubah atau dipindahkan maka akan terjadi redundansi pada proses model)</p>
<i>Activity Automation</i>	<ul style="list-style-type: none"> • Melibatkan <i>systems</i> sebagai <i>actor</i> pada proses model <i>to-be</i> (<i>systems</i> disini dapat dikatakan sebagai <i>actor</i> karena <i>systems</i> akan memutuskan apakah terdapat kendaraan atau tidak untuk memenuhi <i>request</i> yang ada).
<i>Integration</i>	<ul style="list-style-type: none"> • Tidak dapat diterapkan, karena semua <i>actor</i> berasal dari internal perusahaan sendiri, tidak melibatkan <i>supplier</i> ataupun <i>customer</i>.

Setelah melakukan penerapan seperti pada tabel diatas, maka didapatkanlah hasil proses model *to-be* yang seperti berikut ini:



Gambar A.1. Proses model *to-be*

LAMPIRAN B

Lampiran B berisikan Layout wawancara (process discovery - interview)

Pada lampiran ini akan ditampilkan *layout* dari pertanyaan *interview* ketika tahapan *process discovery* dilakukan. Kondisi penulis ketika melakukan *interview* ini adalah berada di ruangan admin supir PT.PERTAMINA EP yang merupakan penanggung-jawab sekaligus *actor* dari sistem reservasi kendaraan yang ada di PT.PERTAMINA EP. berikut pertanyaannya :

2.1 Pertanyaan Pengantar

Pertanyaan pengantar ini bertujuan untuk memastikan jalannya proses secara garis besar serta penanggung-jawab proses ini, berikut pertanyaannya :

- Departemen manakah yang bertanggung jawab atas kendaraan-kendaraan yang ada di PT.PERTAMINA EP ? tepatnya pada sub-departemen apa ?
- Apakah hanya bapak saja (admin supir) yang menanggapi *request* reservasi kendaraan dari seluruh karyawan di PT.PERTAMINA EP ini ?
- Apakah yang akan dilakukan jika kendaraan habis atau tidak tersedia ?
- Apakah terdapat perbedaan “jatah” untuk setiap jabatan ?

2.2 Pertanyaan Teknis

Pertanyaan ini bersifat teknis yang nantinya akan menjadi dasar bagi penulis untuk membuat proses model *as-is* pada tugas akhir kali ini, berikut pertanyaannya :

- Bagaimana cara karyawan melakukan *request* saat ini ?

B-2

- Apa saja yang menjadi data *input* dari karyawan ketika melakukan *request* ?
- Apakah setiap supir akan menunggu karyawan hingga selesai urusannya ? atau supir langsung kembali ke kantor setelah mengantar ?
- Apa saja jenis mobil yang disediakan untuk reservasi kendaraan di PT.PERTAMINA EP ?
- Apa masalah yang sering dihadapi oleh bapak (admin supir) dalam menjalankan pekerjaan ini ?
- Apakah karyawan PT.PERTAMINA EP sering melanggar aturan yang ada ketika melakukan reservasi kendaraan ? berapa besar persentasenya ?

LAMPIRAN C

Lampiran C berisikan *testing* dari aplikasi yang dibuat pada tugas akhir kali ini.

Lampiran ini merupakan lampiran *testing* dimana aplikasi akan diuji coba berdasarkan scenario yang ada, skenario dibuat berdasarkan kemungkinan-kemungkinan yang terjadi dalam aplikasi, berikut pembahasannya :

3.1 Terdapat Kendaraan Yang Tersedia

Pada *testing* ini diasumsikan terdapat kendaraan yang tersedia ketika terjadi *request*, berikut ini adalah urutan *case* serta *actor* yang melakukan kegiatan ketika terdapat kendaraan yang tersedia untuk memenuhi *request* :

Table C.11.1 tabel *case* dan *actor* (*testing 1*)

<i>case</i>	<i>actor</i>
mengisi form	customer
menanggapi form	admin supir
mengisi data supir	admin supir
pre check-in	customer
melakukan check-in kendaraan	customer
mengkonfirmasi check-in	admin supir
update status kendaraan (check-in)	admin supir
melakukan check-out kendaraan	customer
mengkonfirmasi check-out	admin supir
update status kendaraan (check-out)	admin supir

Berikut ini adalah tampilan (*user interface*) dari aplikasi ketika terdapat kendaraan yang tersedia untuk memenuhi *request* :

Bonitasoft

VRS

staff | Logout | Bonita BPM Portal

Mengisi Form

Id karyawan

staff123

Jumlah penumpang *

Tanggal keberangkatan *

2016 July 24

Catatan tambahan *

Tujuan keberangkatan *

PERTAMINA PERSERO
SKK MIGAS
KEMENTERIAN PERMINYAKAN

Jenis keberangkatan *

sekali-antar
antar-jemput

SEND TO ADMIN

Gambar C.1. case "mengisi form"

Bonitasoft

Welcome: admin

Tasks

Cases

Processes

Menanggapi Form

Id karyawan

staff123

Jumlah penumpang

1

Tanggal keberangkatan

2016 July 24

Catatan tambahan

test

Kendaraan available yang dapat digunakan

avanza 1
avanza 2
avanza 3
avanza 4

Tersedia kendaraan

☒

Feedback untuk karyawan

test

CONTINUE

Gambar C.2. case "menanggapi form"

Bonitasoft

Welcome: admin | User | Settings

Tasks

Cases

Processes

Mengisi Data Supir

Id karyawan

Pertumma123

Tersedia kendaraan

0

Kendaraan yang digunakan

avanza 1

Identitas supir *

supir 1 - andung akbar
supir 2 - suhermanto
supir 3 - cahaylen dhar
supir 4 - aditya mayapada

SEND TO CUSTOMER

Gambar C11.3. case "mengisi data supir"

Bonitasoft Welcome: staff User Settings

Tasks Cases Processes

Membaca Feedback

Id karyawan Pertamina123	Tujuan keberangkatan REMENTRIAN PERMIVAHAN
Jumlah penumpang 1	Jenis keberangkatan antar jemput (geng) antar dan pulang dijemput kembali ke pertamina
Tanggal keberangkatan 2016 June 13	
Tersedia kendaraan 1	
Feedback untuk karyawan test tersedia kendaraan	
Identitas supir supir 8 - aditya mayapada	Kendaraan yang digunakan busaka 1
Pesan default silahkan datang 5 menit sebelum keberangkatan untuk konfirmasi check-in	

Done

Gambar C11.4. case "Pre Check-in "

Bonitasoft Welcome: staff User Settings

Tasks Cases Processes

Melakukan Check In kendaraan

Id karyawan Pertamina123	Tujuan keberangkatan REMENTRIAN PERMIVAHAN
Jumlah penumpang 1	Jenis keberangkatan antar jemput (geng) antar dan pulang dijemput kembali ke pertamina
Tanggal keberangkatan 2016 June 13	
Identitas supir supir 8 - aditya mayapada	Kendaraan yang digunakan busaka 1
saya sudah mengambil kunci mobil dan siap berangkat bersama supir	

Check In

Gambar C.5. case "melakukan check-in kendaraan"

Bonitasoft Welcome: admin User Settings

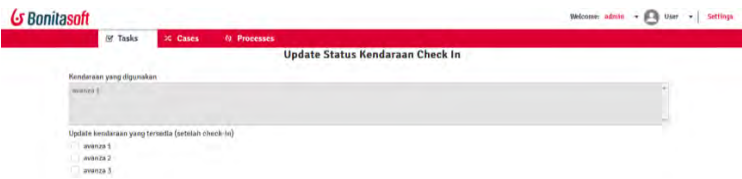
Tasks Cases Processes

Mengkonfirmasi Check In

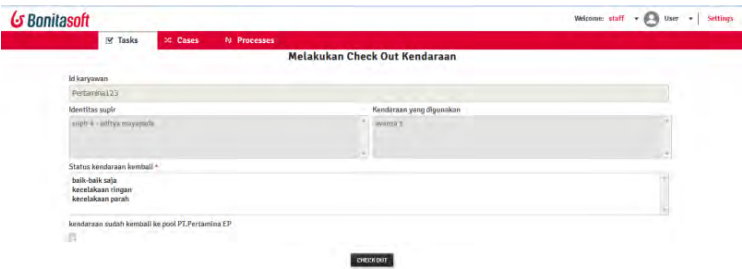
Id karyawan Pertamina123	Tujuan keberangkatan REMENTRIAN PERMIVAHAN
Jumlah penumpang 1	Jenis keberangkatan antar jemput (geng) antar dan pulang dijemput kembali ke
Tanggal keberangkatan 2016 June 13	
Identitas supir supir 8 - aditya mayapada	Kendaraan yang digunakan busaka 1
saya sudah mengambil kunci mobil dan siap berangkat bersama supir	

Confirm

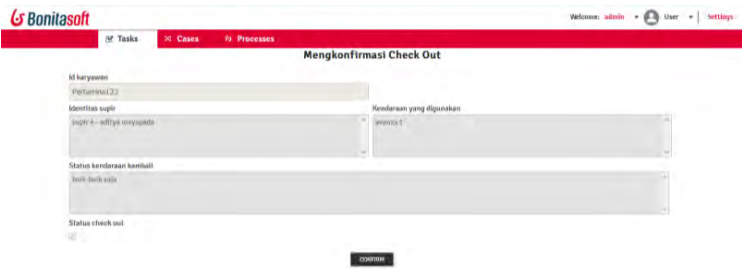
Gambar C11.6. case "mengkonfirmasi check-in"



Gambar C11.7. case "update status kendaraan (check-in)"



Gambar C11.8. case "melakukan check-out kendaraan"



Gambar C11.9. case "mengkonfirmasi check-out"



Gambar C11.10. case "update status kendaraan (check-out)"

3.2 Tidak Terdapat Kendaraan Yang Tersedia

Pada *testing* ini diasumsikan tidak terdapat kendaraan yang tersedia ketika terjadi *request* sehingga admin supir perlu memasukkan *voucher* taksi dan mengirimkannya kepada karyawan (*customer*), berikut ini adalah urutan *case* serta *actor* yang melakukan kegiatan ketika tidak terdapat kendaraan yang tersedia untuk memenuhi *request* :

Table C11.2 tabel *case* dan *actor* (*testing 2*)

<i>case</i>	<i>actor</i>
mengisi <i>form</i>	customer
menanggapi <i>form</i>	admin supir
mengisi kode <i>voucher</i> taksi	admin supir
menerima kode <i>voucher</i> taksi	customer

The screenshot shows a web interface titled "Mengisi Form" within the Bonitasoft VRS environment. The interface includes a header with the Bonitasoft logo, the text "VRS", and user information "staff | Logout | Bonita BP". The form itself contains several input fields: "Id karyawan" with the value "staff123", "Tujuan keberangkatan" with a dropdown menu showing "PERTAMINA PERSERO", "SKK MIGAS", and "KEMENTERIAN PERMINYAKAN", "Jumlah penumpang" which is empty, "Jenis keberangkatan" with a dropdown menu showing "sekali-antar" and "antar-jemput", "Tanggal keberangkatan" with the date "2016 July 24" and a calendar icon, and "Catatan tambahan" which is empty. At the bottom of the form is a button labeled "SEND TO ADMIN".

Gambar C11.11. *case* "mengisi *form*"

Bonitasoft

Welcome: admin

Tasks

Cases

Processes

Menanggapi form

Id karyawan

staff123

Tujuan keberangkatan

PERTAMINA PERSERO

Jumlah penumpang

1

Jenis keberangkatan

sekaligus antar

Tanggal keberangkatan

2016 July 24

Catatan tambahan

test

Tersedia kendaraan

☒

Kendaraan available yang dapat digunakan

avanza 1

avanza 2

avanza 3

avanza 4

Feedback untuk karyawan

test

CONTINUE

Gambar C11.12. case "menanggapi form"

Bonitasoft

Welcome: admin

User

Settings

Tasks

Cases

Processes

Mengisi Voucher Taksi

Id karyawan

permana123

Voucher taksi

SEND TO CUSTOMER

Gambar C11.13. case "mengisi kode voucher taksi"

Bonitasoft

Welcome: staff

User

Settings

Tasks

Cases

Processes

Menerima Kode Voucher Taksi

Id karyawan

permana123

Voucher taksi

voucher123

DONE

Gambar C11.14. case "menerima kode voucher taksi"

BAB VII

KESIMPULAN DAN SARAN

7.1 Kesimpulan

Tugas akhir ini membahas tentang *process analysis*, *process discovery*, *process redesign* dan *process implementation* dari sistem reservasi kendaraan di PT.PERTAMINA EP. Dari penelitian-penelitian yang sudah dilakukan, peneliti menemukan :

- Permodelan *as-is* pada tugas akhir kali ini menggunakan metode *business process management* (BPM). Pada permodelan *as-is* ini ditemukan bahwa terdapat 2 *actor* pada proses model dan terdapat 12 *case* yang ada.
- *Actor* yang terlibat pada proses model *as-is* adalah karyawan-karyawan dari PT.PERTAMINA EP itu sendiri yaitu admin supir (karyawan departemen SCM) dan karyawan lainnya (selain admin supir) yang melakukan *request* kendaraan kantor.
- Implementasi dilakukan menggunakan BonitaSoft yang merupakan salah satu *software* BPM terbesar yang ada sampai saat ini.
- Implementasi menggunakan BonitaSoft akan dilakukan dengan cara memasukka proses model *as-is* yang sudah dirancang, membuat organisasi yang menjadi studi kasus, mendeklarasikan *actor – role – username*, mendeklarasikan *variable* dan melakukan pemetaan *actor* yang sudah dibuat terhadap *username* dan *role* yang ada pada organisasi yang sudah dideklarasikan sebelumnya.
- Implementasi menggunakan BonitaSoft akan mengautomasi proses bisnis yang berjalan pada studi kasus sehingga tercipta aplikasi untuk implementasi proses model yang ada.

7.2 Saran

Dari pelaksanaan penelitian tugas akhir ini dapat diberikan saran untuk penelitian selanjutnya antara lain :

7.2.1. *Leverage*

- Dapat dibuatkan versi *mobile* dari aplikasi ini, sehingga para karyawan PT.PERTAMINA EP dapat lebih mudah dan fleksibel untuk melakukan pemesanan kendaraan kantor.
- Melakukan analisis penambahan jam lembur, jika terdapat supir yang melakukan pekerjaan diluar jam kerja (07.00 – 16.00). Setelah itu jika memang terdapat jam lembur (untuk supir) maka dapat diintegrasikan dengan sistem informasi gaji yang ada di PT.PERTAMINA EP (untuk memberikan notifikasi bahwa supir yang bersangkutan sudah melakukan lembur).
- Menambahkan fitur *management*, yaitu fitur yang menampilkan data dari masing-masing kendaraan yang ada di PT.PERTAMINA EP (data yang dimaksud seperti nomor plat, tahun produksi, jenis bahan bakar, *kilometer*, tanggal terakhir melakukan *service*, dan lain-lain) yang dapat dilihat oleh admin supir, fungsinya adalah agar admin supir dapat mengelola (*management*) kendaraan-kendaraan yang ada di PT.PERTAMINA EP (mengelola disini artinya adalah menjaga atau merawat).

7.2.2. *Escalate*

- Penelitian/analisis penerapan proses reservasi kendaraan ini pada PT.PERTAMINA EP cabang lainnya ataupun organisasi eksternal lainnya.
- Pengembangan aplikasi atau sistem serupa seperti sistem reservasi ruang *meeting* pada suatu perusahaan atau sistem reservasi lainnya (memiliki konsep serupa namun objek reservasinya berbeda).

DAFTAR PUSTAKA

- [1] PT.PERTAMINA EP. (2015, Jan.) Sekilas Perusahaan PT. PERTAMINA EP. [Online]. www.pertaminaep.com
- [2] ARIE SADEWA, "pengembangan website untuk pemesanan kendaraan dan paket wisata secara online (studi kasus : sabila transport)," *pengembangan website untuk pemesanan kendaraan dan paket wisata secara online (studi kasus : sabila transport)*, vol. I, pp. 1-3, May 2012.
- [3] S.Si, M.Kom aris wahyudi setia astuti, "aplikasi reservasi sewa mobil online pada rental barokah berbasis web," *aplikasi reservasi sewa mobil online pada rental barokah berbasis web*, vol. I, pp. 1-5, June 2014.
- [4] marlon dumas, *fundamentals of business process management*, 1st ed. tartu, estonia: springer, 2013.
- [5] Bernardo Nugroho Yahya, Arif Wibisono, Hyerim Bae, and Kwangyeol Ryu, "Bayesian network for finding best combination of performers in BPM envirointment," *Bayesian network for finding best combination of performers in BPM envirointment*, vol. I, p. 1, april 2011.
- [6] Arif Wibisono, Bernardo N, Yahya, and Hyerim Bae, "PERFORMERS SELECTION IN BPM ," *PERFORMERS SELECTION IN BPM* , vol. I, p. 3, 2010.
- [7] Cristian Olariu and Cosmania Carmen Aldea, "Managing processes for Virtual Teams – a BPM approach ," *Managing processes for Virtual Teams – a BPM approach* , vol. 109, pp. 380-384, 2013.
- [8] Amna Shifia Nisafani, Arif Wibisono, Seung Kim, and Hyerim Bae, "Bayesian Selection Rule for Human-Resource Selection in Business Process," *Bayesian Selection Rule for Human-Resource Selection in Business Process*, vol. 17, no. 1, p. 4, Apr. 2014.
- [9] Youness LEMBARET, David CLIN, Jean-Pierre

BOUREY Hui LIU, "Comparison between Collaborative Business Process ," *Comparison between Collaborative Business Process* , pp. 1-6, May 2011.

- [10] JIŘÍ BARTA TOMÁŠ LUDÍK, "Architecture for operational processes improvement in emergency management," *Architecture for operational processes improvement in emergency management*, 2011.

BIODATA PENULIS



Penulis dilahirkan di Jakarta pada tanggal 26 April 1994. Penulis merupakan anak pertama dari tiga bersaudara. Penulis menempuh pendidikan di TK Mardi Yuana Depok, SD Mardi Yuana Depok, SMP Mardi Yuana Depok dan SMA Kolese Gonzaga. Pada tahun 2012 penulis diterima di jurusan Sistem Informasi – Institut Teknologi Sepuluh Nopember (ITS) melalui jalur Mandiri (SMITS) dan terdaftar dengan NRP 5212100137.

Selain kesibukan akademik, penulis juga mengikuti berbagai kegiatan kemahasiswaan dan kepanitian. Tugas akhir yang dipilih penulis di Jurusan Sistem Informasi ini masuk ke dalam bidang minat laboratorium Sistem Enterprise. Penulis dapat menghubungi melalui e-mail joshuasatriawan94@gmail.com.